

NUMERICAL AND EXPERIMENTAL ANALYSIS OF MULTI-STAGE AXIAL
TURBINE PERFORMANCE AT DESIGN AND OFF-DESIGN CONDITIONS

A Dissertation

by

SHERIF ALYKADRY ABDELFATTAH

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY

Chair of Committee,	M.T. Schobeiri
Committee Members,	J.C. Han
	S.C. Lau
	H.C. Chen
Head of Department,	A. Polycrapou

August 2013

Major Subject: Mechanical Engineering

Copyright 2013 Sherif Alykadry Abdelfattah

ABSTRACT

Computational fluid dynamics or CFD is an important tool that is used at various stages in the design of highly complex turbomachinery such as compressor and turbine stages that are used in land and air based power generation units. The ability of CFD to predict the performance characteristics of a specific blade design is challenged by the need to use various turbulence models to simulate turbulent flows as well as transition models to simulate laminar to turbulent transition that can be observed in various turbomachinery designs. Moreover, CFD is based on numerically solving highly complex differential equations, which through the use of a grid to discretize the geometry introduces numerical errors. All these factors combine to challenge CFD's role as a predictor of blade performance. It has been generally found that CFD in its current state of the art is best used to compare between various design points and not as a pure predictor of performances.

In this study the capability of CFD, and turbulence modeling, in turbomachinery based geometry is assessed. Three different blade designs are tested, that include an advanced two-stage turbine blade design, a three stage 2D or cylindrical design and finally a three stage bowed stator and rotor design. All cases were experimentally tested at the Texas A&M university Turbomachinery Performance and Flow Research Laboratory (TPFL). In all cases CFD provided good insights into fundamental turbomachinery flow physics, showing the expected improvement from using 2D cylindrical blades to 3D bowed blade designs in abating the secondary flow effects which are dominant loss generators. However, comparing experimentally measured performance results to numerically predicted shows a clear deficiency, where the CFD overpredicts performance when compared to experimentally obtained data, largely underestimating the various loss mechanisms. In a relative sense, CFD as a tool allows the user to calculate the impact a new feature or change can have on a baseline design. CFD will also provide insight into what are the dominant physics that explain why a change can provide an increase or decrease in performance.

Additionally, as part of this study, one of the main factors that affect the performance of modern turbomachinery is transition from laminar to turbulent flow. Transition is an influential phenomena especially in high pressure turbines, and is sensitive to factors such as upstream incident wake frequency and turbulence intensity. A model experimentally developed, is implemented into a CFD solver and compared to various test results showing greater capability in modeling the affects of reduced frequency on the transition point and transitional flow physics. This model is compared to industry standard models showing favorable prediction performance due to its ability to account for upstream wake effects which most current model are unable to account for.

DEDICATION

To my wife and son....

ACKNOWLEDGMENTS

I would like to take this opportunity to thank Dr. M.T Schobeiri for giving me this opportunity to return to the Turbomachinery Laboratory at Texas A&M University, for his guidance and constant suggestions throughout the course of this study, and his patience with me. I would like to thank Dr. S.C. Lau, Dr. J.C. Han, and Dr. H.C. Chen for agreeing to serve on my committee.

Thanks are due to my friends at the Turbomachinery Lab, Hicham Chibli and Kapil Sharma. Eddie Denk for his help and constant advice. Many thanks go to Francis Dang of the Texas A&M University Supercomputing staff for his help and for providing me a platform to conduct some of these simulations. Thanks are due to the Texas A&M University Office of Graduate Studies, who have been extremely responsive to all my concerns and questions whilst I was in Ohio. I would also like to take an opportunity to thank my colleagues at GE Aviation.

Finally I would like to thank Sherine Jamus my wife for her constant support and for always being there to help me through the tough times. My son Karim, hopefully he'll read this one day. My parents and parents-in-law for all their support and faith.

NOMENCLATURE

h	Specific Enthalpy	$[J/kg]$
H	Total Enthalpy	$[J/kg]$
H_r	Relative Total Enthalpy	$[J/kg]$
L_m	Mixing Length	$[m]$
\dot{m}	Mass flow rate	$[kg/sec]$
M	Mach number	
M_r	Relative Mach number	
N	Machine rotational speed	$[RPM]$
PR	Static pressure ratio	
P	Power	$[W]$
p	Static pressure	$[Pa]$
P	Total pressure	$[Pa]$
P_r	Relative total pressure	$[Pa]$
p_n	Normalized static pressure	
P_n	Normalized total pressure	
P_{rn}	Normalized relative total pressure	
$RANS$	Reynolds Averaged Navier-Stokes	
Re	Reynolds Number	
Re_θ	Momentum Thickness Reynolds Number	
C_d	Dissipation Coefficient	
D	Diameter	$[m]$
r^*	Normalized Span $r^* = \frac{r-r_{hub}}{r_{tip}-r_{hub}}$	
t	Time	$[sec]$
T	Static Temperature	$[K]$
T_t	Total Temperature	$[K]$

T_{tr}	Relative Total Temperature	$[K]$
$URANS$	Unsteady Reynolds Averaged Navier-Stokes	
U	Circumferential Velocity	$[m/s]$
k	Turbulence Kinetic Energy	$[m^2/s^2]$
k	Thermal Conductivity	$[W/m\ K]$
ω	Turbulence Eddy Frequency	$[1/s]$
c	Chord	$[m]$
s	Blade Spacing	$[m]$
u/c_o	Dimensionless Performance Parameter, $u/c_o = \frac{\pi n r_{mid}}{30 \sqrt{\Delta H_{stage}}}$	
V	Absolute velocity	$[m/s]$
W	Relative velocity	$[m/s]$

Greek Symbols

α	Absolute velocity flow angle	$[deg]$
β	Relative velocity flow angle	$[deg]$
η_{t-s}	Total-to-static Efficiency	
η_{t-t}	Total-to-total Efficiency	
δ	Boundary Layer Thickness	$[m]$
δ_1	Displacement Thickness	$[m]$
δ_2, θ	Momentum Thickness	$[m]$
δ_{ij}	Kronecker Delta	
γ	Meridional Velocity Flow Angle	$[deg]$
γ	Intermittency Function	
Γ	Relative Intermittency	
κ	Von Karman Constant $\kappa = 0.41$	
k	Specific Heat Ratio C_p/C_v	
μ	Dynamic Viscosity	$[\frac{kg\ m}{s}]$
μ_t	Turbulent Viscosity	$[\frac{kg\ m}{s}]$
Ω	Vorticity	$[m/s^2]$
ρ	Density	$[kg/m^3]$
σ	Arc Length Ratio	
τ_{ij}	Shear Stress Tensor	$[Pa]$
ε	Turbulence Dissipation	$[m^3/s^2]$
ζ_s	Stator Total pressure loss coefficient	
ζ_r	Rotor Total pressure loss coefficient	

TABLE OF CONTENTS

	Page
ABSTRACT	ii
DEDICATION	iv
ACKNOWLEDGMENTS	v
NOMENCLATURE	vi
TABLE OF CONTENTS	ix
LIST OF FIGURES	xii
LIST OF TABLES	xxii
1. INTRODUCTION	1
2. TURBOMACHINERY FLOW LOSSES	4
2.1 Turbomachinery Flow Angles and Parameters	5
2.2 Profile/Boundary Layer Losses	8
2.3 Trailing Edge Losses	9
2.4 Tip Leakage Losses	9
2.5 Endwall/Secondary Losses	11
2.6 Application of CFD to Turbomachinery	15
2.6.1 Steady-state Modeling of Axial Turbines	15
2.6.2 Unsteady Modeling of Axial Turbines	20
3. TRANSITIONAL FLOW REGIMES	28
3.1 Natural Transition	28
3.2 Separated Flow Transition	29
3.3 Bypass Transition	29
3.4 Transition in Turbomachinery	30
4. RESEARCH OBJECTIVES	33
4.1 Numerical Mesh	33
4.2 Steady State Analysis	34
4.3 Transient Analysis	34
4.4 Transition Modeling in Turbomachinery and Boundary Layer Flows	35
5. COMPUTATIONAL FLUID DYNAMICS AND TURBULENCE MODELING	36
5.1 Reynolds Averaged Navier Stokes (RANS)	39
5.2 Turbulence Modeling	42

5.2.1	Zero Equation Models	42
5.2.2	One-Equation Models	45
5.2.3	Two Equation Models	45
6.	EXPERIMENTAL FACILITY	52
6.1	Compressor and Electric Drive Unit	52
6.2	Multi-Stage Turbine Engine	54
6.3	Inlet Air Heater	54
6.4	Torque Meter and Dynamometer	55
6.5	TPFL Instrumentation	55
6.5.1	Probe Traversing Unit	55
6.5.2	Engine Monitoring Instrumentation	56
6.5.3	Turbine Performance Instrumentation	56
6.6	Data Acquisition	57
7.	TWO-STAGE TURBINE STUDY	59
7.1	Numerical Simulation	62
7.1.1	Mesh Generation	62
7.1.2	Boundary Conditions	63
7.1.3	Turbulence Model	64
7.1.4	Grid Independence	65
7.1.5	Steady (RANS) and Unsteady (URANS) Analyses	66
7.2	Analysis of Numerical Results	70
7.2.1	Seal Leakage	71
7.2.2	Flow Visualization: Design and Off-design Conditions	73
7.2.3	Comparison to Experimental Data	79
7.2.4	Tip Leakage Effects	87
8.	THREE STAGE TURBINE STUDY	93
8.1	Blade Designs and Mesh	93
8.2	Numerical Simulation	97
8.3	RANS Based Results	98
8.3.1	Three Stage Turbine Cylindrical Blade RANS Results	98
8.3.2	3D Bowed Blade RANS Results	109
8.4	Performance Results and 3D Blade Effects	119
8.5	URANS Based Results	123
8.5.1	Cylindrical Blade URANS Results	127
8.5.2	3D Bowed Blade URANS Results	133
8.5.3	3D Unsteady Effects	137
9.	DISCUSSION: 3D TURBOMACHINERY CFD	146
9.1	Modeling Errors	147
9.2	Mixing Planes	148
9.3	Turbulence Modeling	151
9.4	Transition Modeling	152
9.5	Experimental Considerations	154
10.	LAMINAR TO TURBULENT TRANSITION MODELING	156
10.1	Transition Models Summary	156

10.1.1	e^n Method	157
10.1.2	Baldwin-Lomax Algebraic Turbulence Model	157
10.1.3	Warren, Harris and Hassan One Equation Transition Model	157
10.1.4	Linear Combination Transition Model	158
10.1.5	Suzen and Huang Model	158
10.1.6	Walters and Leylek Model	158
10.1.7	$\gamma - Re_\theta$ Model	159
10.1.8	Chakka and Schobeiri Model	162
10.2	Experiments of Laminar to Turbulent Transition	163
10.3	Transition Model Implementation into CFD Solver	169
10.3.1	Curved Plate Facility Grid	170
10.4	Curved Plate Steady Results	171
10.5	Curved Plate Unsteady Results	179
10.6	Blade Cascade Results	186
10.6.1	Blade Cascade: Chakka and Schobeiri Model	186
10.6.2	Blade Cascade: $\gamma - Re_\theta$ Model	190
11.	DISCUSSION:TRANSITION MODELING	194
12.	CONCLUSIONS	196
	REFERENCES	199
	APPENDIX A. TURBINE CFD POSTPROCESS FORTRAN SCRIPTS	211
A.1	Fortran Program to Analyze RANS Based Turbine CFD Data	211
A.2	Fortran Program to Analyze URANS Based Turbine CFD Data	228
	APPENDIX B. CFX FORTRAN USER FUNCTIONS	256
B.1	Intermittency Function	256
B.2	Inlet Velocity/Wakes Function	258

LIST OF FIGURES

FIGURE		Page
1.1	The 66,500 lbf rated GE Aviation GEnx-2B turbofan [1] used in the new Boeing 747-8 Intercontinental.	1
2.1	Enthalpy versus entropy diagram showing expansion in an absolute and relative Frame, obtained from [8].	6
2.2	Velocity vector diagram showing absolute \vec{V} and relative \vec{W} velocity vectors, obtained from [8].	7
2.3	Stator inlet and exit velocity triangles, obtained from [8, 10].	7
2.4	Periodic wake flow exhibited downstream of blade row as illustrated by Schobeiri [8].	8
2.5	(a) Turbine stage with stator and rotor shrouds, and labyrinth seals, (b) Reduction of pressure across a labyrinth by means of dissipation of kinetic energy, obtained from Schobeiri [8].	10
2.6	Turbine cascade flow pattern as measured by Langston et al. [14, 16]. . . .	12
2.7	Turbine cascade flow pattern as measured by Sharma and Butler [15]. . .	12
2.8	Turbine cascade flow pattern as measured by Wang et al. [18].	13
2.9	Compressor mesh used for steady state compressor simulations by [37]. .	18
2.10	Spanwise averaged total pressure plots at various machine locations as shown by [37].	18
2.11	Turbine unsteady simulation mesh used by Pullan in [45].	21
2.12	Time averaged rotor exit yaw angles as calculated by Pullan in [45]. . . .	22
2.13	Predicted total pressure contour at stator exit as calculated by Pullan in [45].	22
3.1	Representation of suction-side, laminar separation showing both reattached (a) and stalled (b), obtained from [75].	29
6.1	Multistage model turbine at Texas A&M University's TPFL as shown in Schobeiri et al. [2].	53

7.1	Two-stage turbine geometry as shown in [2].	60
7.2	Two-stage turbine stator blade geometry details.	61
7.3	Two-stage turbine rotor blade geometry details.	61
7.4	Interstage analysis numerical mesh (mesh4).	63
7.5	Total-to-static and total-to-total efficiencies as a function of node number.	67
7.6	Comparison of circumferentially averaged relative total pressure and total pressure versus normalized span using various mesh densities at (a) station 3, and (b) station 4.	68
7.7	Comparison of circumferentially averaged relative Mach number and absolute Mach number versus normalized span using various mesh densities at (a) station 3, and (b) station 4.	68
7.8	Comparison of circumferentially averaged relative flow angle β and absolute flow angle α versus normalized span using various mesh densities at (a) station 3, and (b) station 4.	69
7.9	Comparison of circumferentially averaged meridional flow angle γ versus normalized span using various mesh densities at (a) station 3, and (b) station 4.	69
7.10	Comparison of circumferentially averaged turbulence kinetic energy k versus normalized span using various mesh densities at (a) station 3, and (b) station 4.	70
7.11	Meridional contour of total pressure in the stator 2 hub seal.	72
7.12	Meridional contour of total pressure in the rotor 2 tip seal.	72
7.13	3D visualization of leakage flow in rotor 2 tip seal.	73
7.14	Two-stage turbine, stage 2 rotor hub and tip vortices for 3000 RPM PR=1.44 case.	74
7.15	Two-stage turbine stage 2 rotor hub and tip vortices for 1800 RPM PR=1.44 case.	74
7.16	Two-stage turbine stage 2 rotor hub and tip vortices for 3000 RPM PR=1.15 case.	75
7.17	Two-stage turbine stage 2 rotor hub and tip vortices for 3000 RPM PR=1.44 case viewing the pressure side.	76
7.18	Two-stage turbine stage 2 rotor hub and tip vortices for 1800 RPM PR=1.44 case viewing the pressure side.	77

7.19	Two-stage turbine stage 2 rotor hub and tip vortices for 3000 RPM PR=1.15 case viewing the pressure side.	77
7.20	Blade loading chart (C_p versus axial position) for the design case (3000 RPM speed and PR=1.4) for 50% immersion (left) and 90% immersion (right).	78
7.21	Blade loading chart (C_p versus axial position) for the off-design case (3000 RPM speed and PR=1.15) for 50% immersion (left) and 90% immersion (right).	78
7.22	Velocity vectors and contours of relative Mach number for the design case (3000 RPM speed and PR=1.4) for 50% immersion (left) and 90% immersion (right).	79
7.23	Velocity vectors and contours of relative Mach number for the off-design case (3000 RPM speed and PR=1.15) for 50% immersion (left) and 90% immersion (right).	79
7.24	Two-stage turbine spanwise averaged static pressure at PR=1.44 and 3000 RPM.	82
7.25	Two-stage turbine spanwise averaged total pressure at PR=1.44 and 3000 RPM.	82
7.26	Two-stage turbine spanwise averaged relative total pressure at PR=1.44 and 3000 RPM.	83
7.27	Two-stage turbine spanwise averaged Mach number at PR=1.44 and 3000 RPM.	83
7.28	Two-stage turbine spanwise averaged relative Mach number at PR=1.44 and 3000 RPM.	84
7.29	Two-stage turbine spanwise averaged absolute flow angle α at PR=1.44 and 3000 RPM.	84
7.30	Two-stage turbine spanwise averaged relative flow angle β at PR=1.44 and 3000 RPM.	85
7.31	Two-stage turbine stator 2 total pressure loss ζ versus span at PR=1.44 and 3000 RPM.	85
7.32	Two-stage turbine rotor 2 total pressure loss ζ versus span at PR=1.44 and 3000 RPM.	86
7.33	Two-stage turbine contours of relative Mach number at station 3 for (a) experimental (b) RANS and (c) averaged URANS at PR=1.44.	86

7.34	Two-stage turbine contours of relative flow angle β at station 3 for (a) experimental (b) RANS and (c) averaged URANS at PR=1.44.	87
7.35	Two-stage turbine contours of normalized total pressure at station 4 for (a) experimental (b) RANS and (c) averaged URANS at PR=1.44.	87
7.36	Two-stage turbine rotor blade mesh using a tip clearance region.	88
7.37	Two stage turbine total-to-static efficiency versus U/Co	90
7.38	Tip leakage streamlines at 1800 RPM.	91
7.39	Tip leakage streamlines at 3000 RPM.	91
7.40	Tip leakage vorticity contours for the 1800 RPM case.	92
7.41	Tip Leakage vorticity contours for the 3000 RPM case.	92
8.1	Cylindrical blade series, showing both stator and rotor profiles and blade exit metal angles.	93
8.2	3D bowed blade rotor profile and bowing angles.	94
8.3	Cylindrical blade series mesh.	95
8.4	3D bowed blade series mesh.	96
8.5	TPFL three stage configuration cross-section, obtained from [91].	97
8.6	Cylindrical blade circumferentially averaged static pressure at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.	99
8.7	Cylindrical blade circumferentially averaged static pressure at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.	99
8.8	Cylindrical blade circumferentially averaged total pressure at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.	100
8.9	Cylindrical blade circumferentially averaged total pressure at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.	100
8.10	Cylindrical blade circumferentially averaged relative total pressure at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.	101
8.11	Cylindrical blade circumferentially averaged relative total pressure at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.	101
8.12	Cylindrical blade circumferentially averaged absolute Mach number at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.	102

8.13	Cylindrical blade circumferentially averaged absolute Mach number at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.	102
8.14	Cylindrical blade circumferentially averaged relative Mach number at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.	103
8.15	Cylindrical blade circumferentially averaged relative absolute Mach number at 2000 RPM for Stations 3, 4 and 5 for the three-stage turbine. . . .	103
8.16	Cylindrical blade experimental and RANS normalized total pressure contour at station 4 and RPM=2600 for the three-stage turbine.	104
8.17	Cylindrical blade experimental and RANS normalized total pressure at station 4 and RPM=2000 for the three-stage turbine.	104
8.18	Cylindrical blade experimental and RANS normalized relative total pressure contour at station 3 and RPM=2600 for the three-stage turbine. . . .	105
8.19	Cylindrical blade experimental and RANS normalized relative total pressure contour at station 3 and RPM=2000 for the three-stage turbine. . . .	105
8.20	Cylindrical blade experimental and RANS Mach contours at station 4 and RPM=2600 for the three-stage turbine.	106
8.21	Cylindrical blade experimental and RANS Mach contours at station 4 and RPM=2000 for the three-stage turbine.	106
8.22	Cylindrical blade experimental and RANS relative Mach contours at station 3 and RPM=2600 for the three-stage turbine.	107
8.23	Cylindrical blade experimental and RANS relative Mach contours at station 3 and RPM=2000 for the three-stage turbine.	107
8.24	3D bowed blade circumferentially averaged Static Pressure at 2600 RPM for Stations 3, 4 and 5 for three-stage turbine.	110
8.25	3D bowed blade circumferentially averaged static pressure at 1800 RPM for stations 3, 4 and 5 for the three-stage turbine.	110
8.26	3D bowed blade circumferentially averaged Total Pressure at 2600 RPM for Stations 3, 4 and 5 for the three-stage turbine.	111
8.27	3D bowed blade circumferentially averaged Total Pressure at 1800 RPM for Stations 3, 4 and 5 for the three-stage turbine.	111
8.28	3D bowed blade circumferentially averaged Relative Total Pressure at 2600 RPM for Stations 3, 4 and 5 for the three-stage turbine.	112
8.29	3D bowed blade circumferentially averaged Relative Total Pressure at 1800 RPM for Stations 3, 4 and 5 for the three-stage turbine.	112

8.30	3D bowed blade circumferentially averaged Absolute Mach Number at 2600 RPM for Stations 3, 4 and 5 for the three-stage turbine.	113
8.31	3D bowed blade circumferentially averaged Absolute Mach Number at 1800 RPM for Stations 3, 4 and 5 for the three-stage turbine.	113
8.32	3D bowed blade circumferentially averaged relative Mach number at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.	114
8.33	3D bowed blade circumferentially averaged relative absolute Mach number at 1800 RPM for stations 3, 4 and 5 for the three-stage turbine.	114
8.34	3D bowed blade experimental and RANS normalized Total Pressure Contour at Station 4 and RPM=2600 for the three-stage turbine.	115
8.35	3D bowed blade experimental and RANS normalized total pressure at station 4 and RPM=1800 for the three-stage turbine.	115
8.36	3D bowed blade experimental and RANS normalized relative total pressure contours at station 3 and RPM=2600 for the three-stage turbine.	116
8.37	3D bowed blade experimental and RANS normalized relative total pressure contour at station 3 and RPM=1800 for the three-stage turbine.	116
8.38	3D bowed blade experimental and RANS Mach contours at station 4 and RPM=2600 for the three-stage turbine.	117
8.39	3D bowed blade experimental and RANS Mach contours at station 4 and RPM=1800 for the three-stage turbine.	117
8.40	3D bowed blade experimental and RANS relative Mach contours at station 3 and RPM=2600 for the three-stage turbine.	118
8.41	3D bowed blade experimental and RANS relative Mach contours at station 3 and RPM=1800 for the three-stage turbine.	118
8.42	Total pressure loss coefficient station4 at 2600 RPM for both cylindrical and 3D bowed blades for the three-stage turbine.	120
8.43	Station 4 entropy contours at 1800 RPM for both cylindrical and 3D bowed blades for the three-stage turbine.	121
8.44	Experimental and RANS based total-to-static efficiency η_{t-s} at varying shaft Speeds for cylindrical blades. Pressure ratio=1.4 for the three-stage turbine.	122
8.45	Experimental and RANS based total-to-static efficiency η_{t-s} at varying shaft speeds for 3D bowed blades. Pressure ratio=1.4 for the three-stage turbine.	123

8.46	RANS root mean square residual errors versus iteration number for 1800 rpm cylindrical blade case, using the SST shear stress transport turbulence model.	126
8.47	URANS root mean square residual error versus blade passage period t/T for 1800 rpm cylindrical blade case with the SST shear stress transport turbulence model.	127
8.48	Temporally and circumferentially averaged total pressure at station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	128
8.49	Temporally and circumferentially averaged relative total pressure at station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	129
8.50	Temporally and circumferentially averaged static pressure at station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	129
8.51	Temporally and circumferentially averaged Mach number at Station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	130
8.52	Temporally and circumferentially averaged relative Mach number at Station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	130
8.53	Temporally and circumferentially URANS and RANS based rotor and stator total pressure loss coefficient for the cylindrical blade at 1800 RPM PR=1.4 for the three-stage turbine.	131
8.54	Temporally averaged URANS total pressure contours at stations 3 and 4 versus corresponding RANS data at N=1800 RPM and PR=1.4 for the three-stage turbine.	131
8.55	Temporally averaged URANS entropy [$kJ/kg\ K$] contours at stations 3 and 4 versus corresponding RANS data at N=1800 RPM and PR=1.4 for the three-stage turbine.	132
8.56	Cylindrical blade URANS total-to-static efficiency versus blade passage time for 1800 RPM.	132
8.57	Cylindrical blade URANS massflow [kg/s] versus blade passage time for 1800 RPM.	133
8.58	Temporally and circumferentially averaged relative total pressure at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	134

8.59	Temporally and circumferentially averaged static pressure at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	134
8.60	Temporally and circumferentially averaged Mach number at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	135
8.61	Temporally and circumferentially averaged relative Mach number at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.	135
8.62	Temporally and circumferentially averaged URANS and RANS based rotor and stator total pressure loss coefficient for the 3D bowed blade at 1800 RPM PR=1.4 for the three-stage turbine.	136
8.63	3D Bowed Blade temporally averaged URANS total pressure contours at stations 3 and 4 versus corresponding RANS data at N=1800 RPM and PR=1.4 for the three-stage turbine.	136
8.64	3D bowed blade temporally averaged URANS entropy [$kJ/kg\ K$] contours at stations 3 and 4 versus corresponding RANS data at N=1800 RPM and PR=1.4 for the three-stage turbine.	137
8.65	Cylindrical blade 1800 RPM mid-span entropy contours at various blade passage instants (rotation of rotor blades not shown).	138
8.66	3D bowed blade 1800 RPM mid-span entropy contours at various blade passage instants (rotation of rotor blades not shown).	139
8.67	Cylindrical stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (suction surface view).	140
8.68	3D bowed stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (suction surface view).	141
8.69	Cylindrical stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (pressure surface view).	142
8.70	3D bowed stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (pressure surface view).	143
8.71	Cylindrical stator 2 blade 1800 RPM entropy contours at various axial positions, for various time steps	144
8.72	3D bowed stator 2 blade 1800 RPM entropy contours at various axial positions, for various time steps.	145

9.1	Mixing plane averaging applied to the density at the interface between the two-stage design's stator 2 and rotor 2.	149
9.2	Mixing plane averaging applied to the turbulence kinetic energy at the interface between the two-stage design's stator 2 and rotor 2.	149
9.3	Mixing plane averaging applied to the turbulence eddy frequency at the interface between the two-stage design's stator 2 and rotor 2.	150
9.4	Mixing plane averaging applied to the turbulence eddy dissipation at the interface between the two-stage design's stator 2 and rotor 2.	150
9.5	Mixing plane averaging applied to the temperature (enthalpy) at the interface between the two-stage design's stator 2 and rotor 2.	151
9.6	Contours of intermittency at the mid-span point for the three-stage bowed blade design for both the high and low speed case.	153
10.1	Relative intermittency Γ as a function of non-dimensionalized lateral coordinate for $\Omega = 3.443$, obtained from Chakka and Schobeiri [53].	163
10.2	Curved plate test section of Chakka and Schobeiri [53] with 1. Traversing system, 2. Nozzle, 3. Wake Generator, 4. Electric motor, 5. Convex wall, 6. Concave wall, 7. Hot-wire probe, 8. Plexiglass wall, 9. Curved plate, 10. Small Vernier and 11. Large Vernier.	165
10.3	Maximum, minimum, and time-averaged intermittency distributions as a function of axial Reynolds number for $\Omega = 1.033$ (3 rods) at $y = 0.1$ mm, obtained from Chakka and Schobeiri [53].	167
10.4	Averaged intermittency γ for the steady curved plate case $\Omega = 0$, obtained from Chakka and Schobeiri [53].	167
10.5	Numerical grid for plate facility transition simulations (coarse mesh with 38,000 elements shown).	171
10.6	Momentum thickness θ as a function of axial Reynolds number at curved plate wall.	173
10.7	Reynolds number momentum thickness θ as a function of axial Reynolds number at the curved plate wall.	173
10.8	Turbulence intermittency γ as a function of axial Reynolds number at the curved plate wall.	174
10.9	Stanton number as a function of axial Reynolds number at the curved plate wall.	175
10.10	Contours of intermittency γ for steady curved plate case.	177

10.11	Contours of velocity for steady case using different transition models for curved plate case.	178
10.12	Unsteady intermittency for $\Omega = 3.78$	181
10.13	Unsteady intermittency for $\Omega = 0.77$	181
10.14	Unsteady Stanton number for $\Omega = 3.78$	182
10.15	Unsteady Stanton number for $\Omega = 0.77$	182
10.16	Temporally averaged intermittency.	183
10.17	Temporally averaged wall shear.	183
10.18	Contours of intermittency at different time phases for $\Omega = 0.77$	184
10.19	Contours of absolute velocity at different time phases for $\Omega = 0.77$	185
10.20	Numerical mesh used for blade cascade simulations.	186
10.21	Displacement thickness δ^* at the pressure and suction surfaces, as predicted by the Chakka and Schobeiri model.	187
10.22	Momentum thickness θ at the pressure and suction surfaces as predicted by the Chakka and Schobeiri model.	188
10.23	Intermittency function γ applied at the pressure and suction surfaces using the Chakka and Schobeiri model.	188
10.24	Momentum thickness Reynolds number Re_θ at the pressure and suction surfaces as predicted by the Chakka and Schobeiri model.	189
10.25	Heat transfer coefficient h at the pressure and suction surfaces as predicted by the Chakka and Schobeiri model.	189
10.26	Displacement thickness δ^* at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.	191
10.27	Momentum thickness θ at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.	191
10.28	Intermittency function γ applied at the pressure and suction surfaces using the $\gamma - Re_\theta$ model.	192
10.29	Momentum thickness Reynolds number Re_θ at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.	192
10.30	Heat transfer coefficient h at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.	193

LIST OF TABLES

TABLE		Page
5.1	Coefficients used in the standard $k-\varepsilon$ model.	48
5.2	Coefficients used in the $k-\omega$ turbulence model.	49
5.3	Coefficients used in the SST turbulence model.	51
6.1	Turbine research facility specifications.	54
7.1	Two-stage turbine research facility specifications.	59
7.2	Mesh densities used in the two-stage turbine numerical analyses.	64
7.3	Two-stage turbine boundary conditions for 1800 rpm and 2600 rpm at a pressure ratio of 1.38.	65
8.1	3-stage cylindrical and bowed blade turbine specifications.	96
8.2	Boundary conditions for 1800 RPM and 2600 RPM cases.	97
8.3	Unsteady boundary conditions for 1800 RPM and 2600 RPM cases.	126
8.4	Total-to-static efficiency comparison for cylindrical and 3D bowed blades for the three-stage turbine 1800 RPM case.	142
10.1	Intermittency gamma (γ) equation constants.	161
10.2	Momentum thickness Reynolds number Re_θ equation constants.	162
10.3	Intermittency correlation constants, as listed in Chakka and Schobeiri [53].	169
10.4	Unsteady plate boundary conditions.	179
10.5	Blade cascade boundary condition for RANS simulation.	187

1. INTRODUCTION

Turbomachinery play an essential role in today's world, providing energy for homes through steam turbines as well as gas turbines, power for aircraft such as the GEnx-2B shown in figure 1.1, and forced induction (turbocharging) for both Otto and Diesel engines. The main challenge turbomachinery design engineers are confronted with, is speeding up the design process, through increased simulation and less experimental testing. One tool that is being used recently, to simulate various turbomachinery is computational fluid dynamics (CFD). The importance of understanding the role CFD plays in the turbomachinery design process, how its results compares to experimentally obtained performance and flow visualization must be stressed. CFD is used to model a machine that exhibits a complex three dimensional flow. The use of numerical schemes, turbulence models and various other simplifications of the physical flow in CFD, result in differences between CFD simulations and experimentally obtained data.

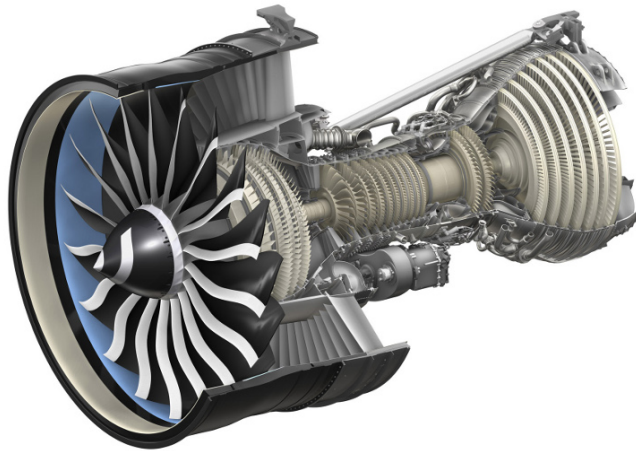


Figure 1.1: The 66,500 lbf rated GE Aviation GEnx-2B turbofan [1] used in the new Boeing 747-8 Intercontinental.

Design of turbomachinery components such as compressors and turbines have advanced

significantly with the introduction of CFD. A turbomachinery designer is now capable of running a steady-state analysis of a multistage machine on an average desktop based computer with a relatively coarse numerical grid describing the geometry. The important question becomes how reliable and trustworthy are results that can be obtained through CFD analysis of 3D turbomachinery? Especially when compared to experimental data. The main source of errors that are exhibited in CFD discussed in detail by Schobeiri et al. [2, 3] as well as Denton [4] and discussed in this study include:

- Numerical errors due to numerical scheme approximations.
- Modeling errors, such as the use of turbulence modeling to model the complex and chaotic nature of turbulence.
- Unknown boundary conditions, such as inlet pressure or temperature profiles.
- Unknown geometry such as hot geometry versus cold geometry as well tip clearances.
- Assumption of steady flow when using a steady simulation to model a multistage turbomachine

Although the numerical techniques and available turbulence modeling have advanced in the past two decades, the usage of CFD in turbomachinery needs to be approached with care. The question that needs to be asked is, can CFD be used in a priori to predict a specific design's performance and in the process match an accurately taken experimental test of that machine? The answer to that question is, on most occasions a resounding no. As shown by Schobeiri et al. [2, 3], and various other authors [5, 6, 7, 4], CFD continues to generally struggle to predict for example, a multistage turbine's performance at both design or nominal operating conditions as well as off-design cases. Turbomachinery flows are highly unsteady, three-dimensional in nature due to regions of near wall interaction, and provide a large challenge to any flow or performance prediction techniques. In a low pressure turbine (LPT) Reynolds numbers as well as other factors including, inlet turbulence, upstream wake reduced frequency determine whether the flow is laminar, fully

turbulent or transitional, this becomes an added challenge for any solver and such physics require models to capture their effects on turbomachinery performance and flow physics.

In this dissertation, the use of CFD to predict the performance of various axial turbine designs is explored, with some emphasis placed on the question of transition modeling in turbomachinery. First a discussion of turbomachinery losses sources, transition flow modeling and prior literature on the use of CFD to model turbomachinery is presented.

2. TURBOMACHINERY FLOW LOSSES

Increased energy demands have placed a greater emphasis on the need for a more and reliant turbine for both power generation and aviation requirements. Computational Fluid Dynamics (CFD) have continued to play a greater role in the turbomachinery industry as a major design tool, both for the actual machine design as well as machine performance evaluation. Even though there have been significant forward strides in technology and knowledge behind the various numerical solving techniques and turbulence closures over the past 30 years, there are still various challenges to simulating the notoriously highly complex flow fields that occur in modern turbomachinery. Some of the main factors that pose the greatest challenge to overcome include:

- Three dimensional secondary flow effects that have a big influence on the passage flow field for both compressor and turbines, which is exacerbated by the decreasing aspect ratio of modern designs
- Large 3-D separation regions are observed in compressor sections
- Transitional flow regimes that are observed in engine turbine sections
- Inherent flow unsteadiness and free stream turbulence especially in off-design running conditions

This section provides a brief overview over the various forms of losses exhibited in turbomachinery flows. As explained in detail by Schobeiri [8] turbomachinery losses can be divided into end wall or secondary losses, profile losses and leakage losses. Profile loss is considered loss effects due to boundary layer development on suction and pressure surfaces of the blade in question. Analyses for profile loss often assume a primarily two-dimensional flow, and losses at the trailing edge are often included into the profile loss category. End wall losses are commonly referred to as secondary flow losses and arise due to secondary flows generated as the annulus boundary layers are convected into the blade

passage. Tip leakage losses arise from leakage type flows across the blade tip for a rotor or stator hub, and depend on whether the blades utilize shrouds or do not. Leakage flows as stated by [9] interact strongly with the secondary flow patterns in the passage, thus it is generally considered that there is a large degree interplay between the various types of losses encountered in turbomachinery. Additionally the author proposes one to think of turbomachinery loss in terms of entropy increase, stating that entropy is a convenient conceptual measure of loss as it is not reference frame dependent as total pressure or total enthalpy is. However, entropy is difficult to use as a measure of loss as it may not be measured directly, but must be obtained by measuring other properties.

In this chapter a literature review of all the relevant research involving turbomachinery modeling using CFD as well the latest in transition modeling is presented.

2.1 Turbomachinery Flow Angles and Parameters

Prior to delving further into a discussion on the various types of flow losses in turbomachinery, it is of use to introduce to the reader various terms associated with turbomachinery which will be extensively used in this thesis. The concept of total pressure and relative total pressure can be illustrated using an H-S diagram as shown in figure 2.1, total pressure in the absolute and relative frame of reference is defined as:

$$P = p + \frac{\rho}{2}V^2 \quad (2.1)$$

V is the velocity magnitude in the absolute reference frame, the relative total pressure is defined as:

$$P_r = p + \frac{\rho}{2}(W^2 - U^2) \quad (2.2)$$

W is the velocity magnitude in the relative reference frame, and U is the rotational velocity. Similarly the total enthalpy is defined as:

$$H = h + \frac{V^2}{2} \quad (2.3)$$

And the relative total enthalpy defined as:

$$H_r = h + \frac{1}{2} (W^2 - U^2) \quad (2.4)$$

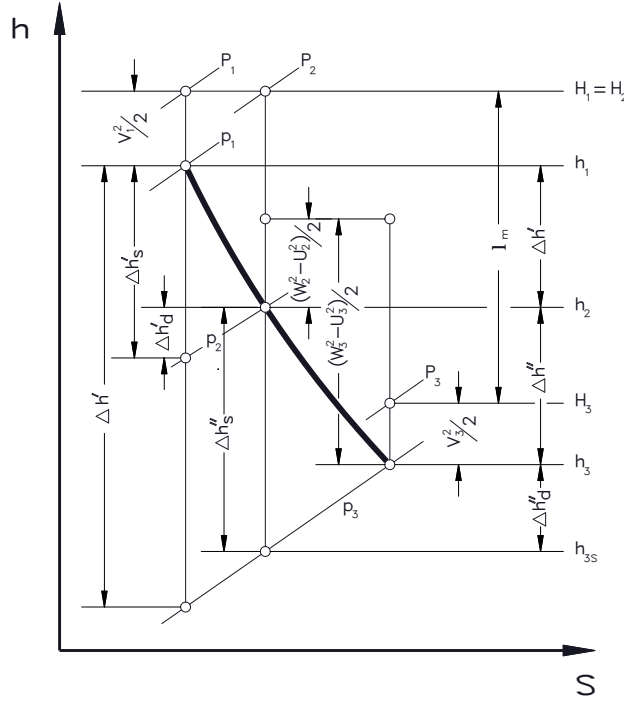


Figure 2.1: Enthalpy versus entropy diagram showing expansion in an absolute and relative Frame, obtained from [8].

Velocity diagrams , or velocity triangles can be constructed to visualize the relative, and absolute velocity vectors as well as the angles associated with the absolute and relative velocity vectors. A velocity triangle is shown in figure 2.2, the absolute angle α is the angle between the absolute velocity vector \vec{V} and the rotational component (horizontal), whilst the relative velocity angle β is the angle between the relative velocity vector \vec{W} and the rotational component (horizontal).

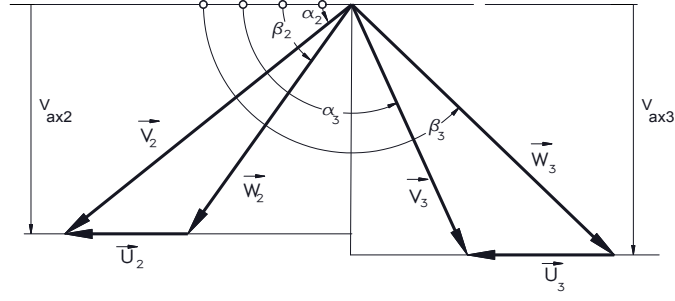


Figure 2.2: Velocity vector diagram showing absolute \vec{V} and relative \vec{W} velocity vectors, obtained from [8].

Figure 2.3 shows the inlet and exit velocity triangles, superimposed onto a blade-to-blade view of a turbine stator blade row.

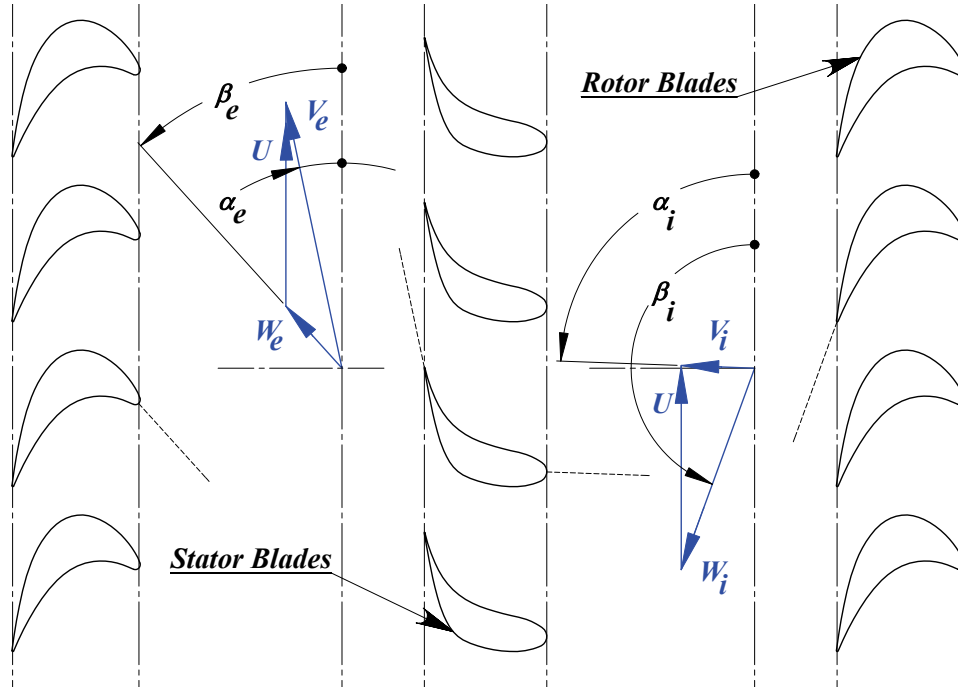


Figure 2.3: Stator inlet and exit velocity triangles, obtained from [8, 10].

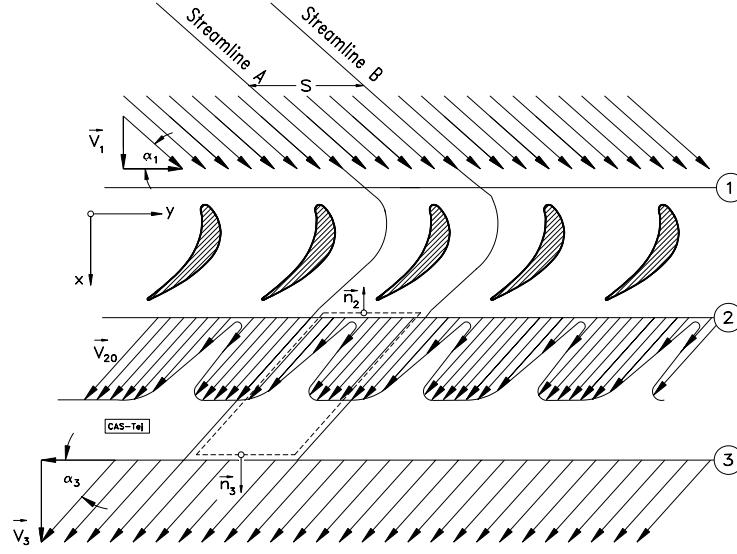


Figure 2.4: Periodic wake flow exhibited downstream of blade row as illustrated by Schobeiri [8].

2.2 Profile/Boundary Layer Losses

With existence of various shear stresses at blade walls due to the non-slip condition, boundary layer developments are exhibited on both pressure and suction surfaces of a blade. In addition to the boundary layer development across the blade length, at the trailing edge of a blade, a specific periodic wake flow is observed downstream of a blade row due to the profile loss effects combined by a trailing edge deficit as highlighted by Schobeiri [8] and shown in Figure 2.4. The shear forces caused by the wake flow present an overall drag force on the blade that in turn reduce the total pressure. The total pressure drop posed in [8] is defined in equation (2.5) as:

$$\zeta = \frac{P_{t1} - P_{t2}}{\frac{1}{2}\rho V^2} \quad (2.5)$$

Based on the derivation by Pfeil, as stated in [8] the blade profile loss coefficient may be calculated using:

$$\zeta = \epsilon_{optimum\ single} \left[1 + 0.25 \left(\frac{c}{s} \right)_{optimum}^3 \right] C_L \left(\frac{c}{s} \right) \frac{1}{\sin \alpha_\infty} \quad (2.6)$$

Equation (2.6) clearly shows that the profile is dependent on several factors that include, the Reynolds number, Mach number effects, blade geometry parameters such as solidity and maximum blade thickness-chord ratio, and finally the surface roughness plays a large part in the blade profile loss. The loss brought about as a result of entropy production can be converted into a dimensionless dissipation coefficient, as described initially by Schlichting [11] and stated as:

$$C_d = \frac{T\dot{S}_a}{\rho V_{delta}^3} \quad (2.7)$$

Schlichting [11] gave the following equation for the dissipation coefficient:

$$C_d = 0.0056 Re_\theta^{-1/6} \quad (2.8)$$

Profile losses are directly proportional to the momentum thickness which, as stated by Schobeiri [8], is inversely proportional to the Reynolds number.

2.3 Trailing Edge Losses

As mentioned earlier trailing edge losses combined with the profile losses create an overall drag force on the blade, and result in a total pressure loss, an increase in entropy and a resultant decrease in blade aerodynamic efficiency. As per Denton [9] measured losses behind a low speed turbine blade, have shown that about a third of the loss is generated behind the trailing edge, showing how significant an effect the trailing edge velocity deficit can have on the overall blade loss. An excellent analysis of trailing edge losses in compressors and turbine is presented by Schobeiri [8], where the author begins with the continuity equation for a blade cascade, and provides an analysis of the underlying physics behind trailing edge losses. The added loss due to trailing edge thickness is shown to significantly increase, as expected with the thickness of the blade trailing edge.

2.4 Tip Leakage Losses

Tip leakage flows occur due to the presence of seals at hub or tip locations for both stator and rotor blades, and the presence of shrouds. The main effect of the such passages is to

influence the massflow in the main blade passage. In the case of compressors, for shrouded blades, the leakage flow would run over the shrouds thereby for an constant overall machine massflow rate, the leakage flow will increase the main blade passage massflow resulting in an increase in work input and a resultant decrease in pressure rise. In the case of shrouded turbine blades, leakage flow from upstream to downstream will result in a drop in blade work and pressure drop over the blade row. Entropy creation due to tip leakage flows is mainly involved with the mixing effects that take place between the leakage flow and the main blade passage flow. This is illustrated in figure 2.5. In a paper by Porreca et al. [12]

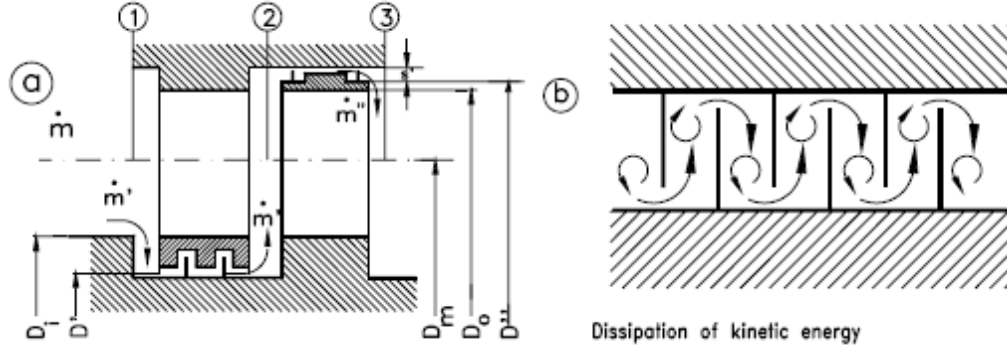


Figure 2.5: (a) Turbine stage with stator and rotor shrouds, and labyrinth seals, (b) Reduction of pressure across a labyrinth by means of dissipation of kinetic energy, obtained from Schobeiri [8].

both an experimental and numerical (CFD) investigation was performed on various shroud arrangements. The experimental data was taken using a Fast Response Aerodynamic Probe (FRAP), the authors found good agreement between CFD results and experimental results for the end wall static pressure contour. The authors show that both experimental results and simulations conclude that the shroud and labyrinth path in the shroud seal can have a significant effect on the overall engine performance. For the partially shrouded case the authors note a large deviation between measured flow angles and blade tip angles in the tip region, and a superior performance for the fully shrouded blade versus the partially

shrouded blade. Adami et al [13] conducted a similar study using a transient CFD analysis to analyze the end wall leakage interaction in low-aspect turbine blading configuration and similarly to Porreca et al. [12], the authors noticed a large effect of cavity flows on the secondary flow field in the blade main passage. Using a minor modification in the shroud geometry created a significant change on the leakage flow ejection and the blockage observed by the mainstream flow field.

2.5 Endwall/Secondary Losses

Secondary flows or endwall flows constitute some of the most commonplace three dimensional flow patterns that are encountered in the generation of electrical and motive power [14]. Sharma and Butler [15] further state that losses due to the endwall or secondary flows account for 30 to 50% of the total aerodynamic loss in a rotor or stator row. The earliest investigation of the secondary flow field was conducted by Langston et al. [16] who used a large scale, low aspect ratio turbine cascade to visualize the flow through a turbine passage. Figure 2.6 shows the resulting measured flow field in the low aspect turbine cascade employed by Langston et al. [16], as shown, at the endwall of the cascade the inlet boundary layer separates at a so-called saddle point and forms a horseshoe vortex. This vortex is then evolved into a passage vortex and it is fed by the endwall flow from the pressure surface of one blade to the suction surface of the adjacent blade. Moreover, the authors noticed the existence of a counter vortex that rotates in an opposite rotational sense to the main passage vortex. The flow pattern shown was also confirmed via the use of smoke flow visualization as shown in [17]. Langston et al. [16] verify the loss of lift effect imparted by the generation of the passage vortex and the counter vortex. Moreover, the authors observe that the net effect of three-dimensional secondary flow patterns is to decrease the area between the suction and pressure side, pressure coefficient distribution, this area is a measure of the net lift, and as such the secondary flows create a loss of turbine work and a net decrease in aerodynamic efficiency.

A modified flow pattern to the one measured by Langston et al. [16] was introduced by Sharma and Butler [15], where the counter vortex was shown to wrap around the passage

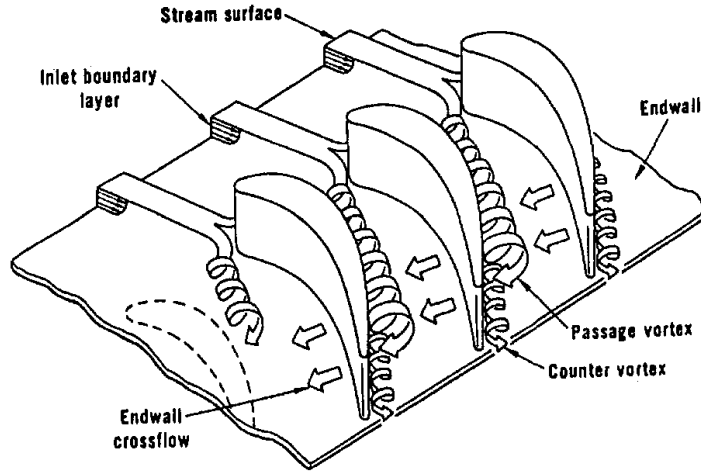


Figure 2.6: Turbine cascade flow pattern as measured by Langston et al. [14, 16].

vortex as shown in Figure 2.7, the authors were able to create a semi-empirical formula to predict turbine losses based on their experimental observations.

Using a laser light technique and multiple smoke wires, Wang et al. [18] measured a multivortex pattern in a turbine cascade. They ran their cascade at a very low Reynolds number and were able to notice a periodically varying horseshoe vortex pattern as with

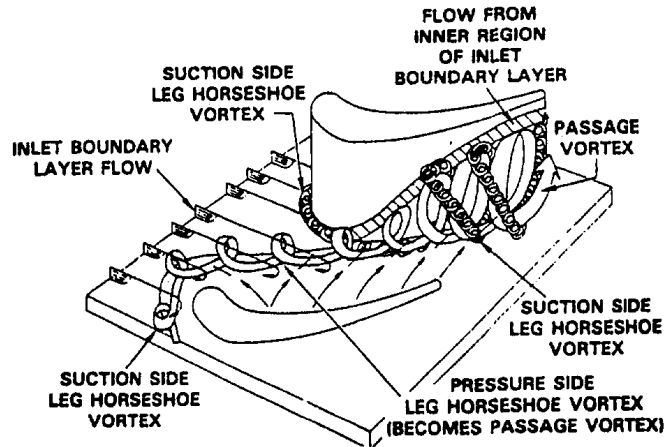


Figure 2.7: Turbine cascade flow pattern as measured by Sharma and Butler [15].

nessed by Baker [19] who used a endwall mounted cylinder run under laminar flow conditions. The measured flow pattern observed by Wang et al. [18] is shown in Figure 2.8.

Eckerle and Langston [20] studied the formation of a horseshoe vortex around a cylindrical object using a wind tunnel. They found that a symmetrical saddle point was observed (which occurred at a location of adverse pressure gradient) upstream of the cylinder. This confirms the statement by Langston [14], that through experimentation it is found that separation at saddle point is not dependent on blade radius of curvature at leading edge but is dependent on the pressure field that is created by the airfoil shape. Eckerle and Awad [21] carried out more detailed investigations of horseshoe vortex formation over a cylinder, and were able to correlate their data using a non-dimensional parameter they termed E which they defined as:

$$E = (Re_D)^{\frac{1}{3}} \left(\frac{D}{\delta^*} \right) \quad (2.9)$$

Re_D is the Reynolds number, D is the cylinder diameter and δ_1 is the displacement thick-

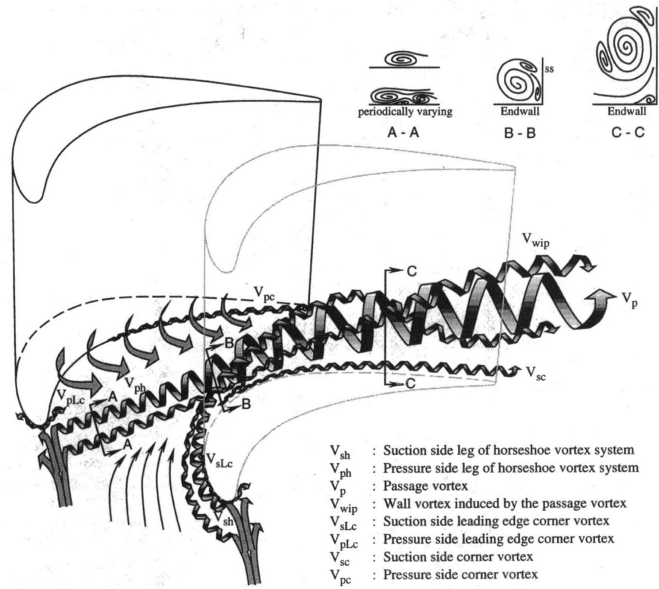


Figure 2.8: Turbine cascade flow pattern as measured by Wang et al. [18].

ness of the boundary layer at the position of the leading edge of the cylinder. Eckerle and Awad [21] showed that for $E > 1000$ no swirling was noticed in a plane upstream of the cylinder, rather the separating boundary layer appeared at an angular distance from the plane of symmetry. For $E < 1000$ the swirling flow field was observed at the plane of symmetry. The correlation obtained by [21], shown in equation (2.9) formed a better way for a turbine designer to predict the saddle point flow behavior in a turbine. The use of CFD codes to predict secondary flows in turbomachinery has been met with limited success, Hah [22] using a modified Reynolds stress model for turbulence closure, was the first to develop a code that managed to predict the majority of the secondary flow effects, when compared to the experimental data put forth by Langston et al [14]. Predicted aerodynamic loss showed good agreement with the trends observed experimentally in [14], but calculated loss coefficients were shown to be higher by 20-30% at the downstream location of the trailing edge. Dorney and Davis [23] were more successful in predicting the effects of secondary flow losses, obtaining an agreement within 2% of the measured value, they used a modified two-layer Baldwin-Lomax algebraic turbulence closure model.

A similar study was performed by Hartland et al. [24] where a simulation of a turbine cascade was performed and compared to experimental data obtained by Gregory-Smith and Cleak [25]. The CFD results quoted by [24] were shown to over-predict losses by as much as 75%. Moreover the code used predicted the planar endwall to be the smaller loss producer, when the experimental data showed an opposite trend. The large discrepancy between measured and calculated losses was attributed by the authors to sub par turbulence model performance. Langston [14] postulates in addition to CFD errors brought about from the turbulence models, the CFD codes are not able to model the highly skewed boundary layer on the endwall, which is observed by Langston [14] to be very thin and laminar. Strongly accelerated laminar flows can give a much higher local friction coefficient when compared to non-accelerated cases.

2.6 Application of CFD to Turbomachinery

Denton [26] was probably the first to present an Euler solver for 2D and 3D turbomachinery based applications, since then and with the rapid advance in computer computational speed over the past 30 years, several commercial finite volume and finite element solvers have been introduced to the market, in addition to various NASA based and academic research codes. Moreover, turbulence modeling has achieved significant strides and the most widely used models in turbomachinery design include the two-equation based $k - \varepsilon$ and the $k - \omega$ models. As stated by Adamczyk [27] there is a small scale chaotic unsteadiness, which is imparted by turbulence in turbomachinery flow, and that is due to this unsteady random flow that a turbomachine is able to either impart or extract energy from a flow. Moreover the author in [27], adds, there is large range of length and time scales in a turbomachine model, which adds a greater degree of difficulty for modeling turbomachinery flows with enough fidelity.

According to Hall [28], prediction schemes for three-dimensional flows through multi-stage turbomachinery can be generally classified into one of three categories [28]:

- Steady flow 'mixing plane' analyses
- Average-passage based flow analyses
- Time-dependent flow analyses or URANS (Unsteady Reynolds Averaged Navier-Stokes) based analysis

2.6.1 Steady-state Modeling of Axial Turbines

The mixing plane analysis was put forth by Denton [26], where the data at the various blade row interface is averaged and then used as boundary conditions for the subsequent downstream rows. This technique ignores all unsteady effects, the average passage based flow analysis by [29] attempts to model some of the unsteady effects via a viscous force analogy and use them for a steady analysis, and lastly the time-dependent flow analysis are fully coupled transient cases which do not use any interface modeling but are very

expensive in terms of computer resources. Some of the simulations available in literature will be examined in this section.

Denton [30] used a three dimensional viscous solver to model a multistage axial turbine. A steady-state analysis was performed using the explicit finite-volume solver, and a novel inter-row mixing model was introduced. Prior to the introduction of the mixing model a simple averaging of upstream flow quantities was applied to the downstream plane and was shown to be deficient and provide unrealistic leading edge blade loading. The model extrapolated the upstream circumferential variation of fluxes whilst maintaining conservation of flow parameters. Dawes [31] used a three-dimensional Navier-Stokes solver to model various secondary flow effects and blade row interaction. The solver used is similar to the code developed by [30], and was applied to a multistage transonic compressor case and a single stage reaction turbine, showing better agreement with experimental data when compared to isolated or cascade studies of the blades modeled.

Adamczyk et al. [29] were able to model a low aspect ratio turbine stage using a three dimensional average-passage and axisymmetric averaging technique, results obtained showed good prediction of blade loading at various span points and the locations of secondary vortices along the radial direction. Mulac and Adamczyk [32] used a parallelized version of the code used in [29] to model a multistage compressor obtaining good stage to stage pressure ratio predictions whilst over-predicting the stage efficiency values. The solver was shown to be slightly deficient in predicting total pressure at the endwall regions. Adamczyk [33] reviews his average-passage model comparing it to various other numerical techniques available to the turbomachinery designer. In the average passage model, average passage stress and total enthalpy flux account in part for the effect of momentum transport and total enthalpy transport attributed to surrounding blade rows. According to Adamczyk [33] there are three components to variables stated, the first due to nondeterministic flow field which accounts for diffusion of total enthalpy and momentum attributed to unsteady processors not linked to shaft rotation, the second due to unsteady deterministic flow which is due to processes linked to shaft speed and are responsible for radial

distribution of total temperature and lastly, the third due to average passage flow field of blade rows other than passage of interest which is associated to unsteady deterministic flow and is responsible for the recovery of the wake mixing loss in compressors, flow blockage and spanwise redistribution of momentum. Chima [34] uses the average passage model of Adamczyk [33] to model the space shuttle main engine fuel turbine showing a favorable comparison between the SWIFT code data and experimental measurements of span-average Stanton numbers. However the author notes some of the shortcomings of the average passage model, which include the inability to account for wake mixing and migration, acoustic interaction, and other unsteady effects.

A three-dimensional finite difference code for turbomachinery applications was developed by Kunz and Lakhminarayana [35] utilizing a $k - \varepsilon$ model and was able to accurately predict various turbine flow features when compared to experimental data. Dawes [36] presented a unstructured solver that also utilizes a $k - \varepsilon$ model provides greater mesh flexibility which enables one to capture more sophisticated geometry. More recently Gerolymos and Hanisch [5] presented a steady-state model of a four stage axial turbine at off-design and design conditions. An in-house code was used that utilized a Favre Reynolds averaged Navier-Stokes (RANS) scheme and a two-equation Launder Sharma $k - \varepsilon$ turbulence model was employed to the implicit upwind solver. The mixing planes between rows used a phantom node technique which creates an overlapping region in between blade rows. The model was found to accurately predict the stage pressure ratios but over-predicted the machine efficiency when compared to corresponding experimental data. Moreover, the model was found to over-predict the circumferentially averaged radial total pressure at various axial positions, at the low massflow condition the discrepancy between numerical and experimental data increased. Gerolymos et al. [6] and Gerolymos and Vallet [37] were able to improve their results through the use of an advanced Reynolds Stress Model (RSM) which uses 7 equations to achieve turbulence closure, 6 equations for the six components of the symmetric Reynolds-stress tensor and a seventh equation to determine scale. The authors showed improved prediction performance when compared to a two equation baseline $k - \varepsilon$

model [5], the mesh and some of the results obtained by Gerolymos and Vallet [37] are shown in Figures 2.9 and 2.10.

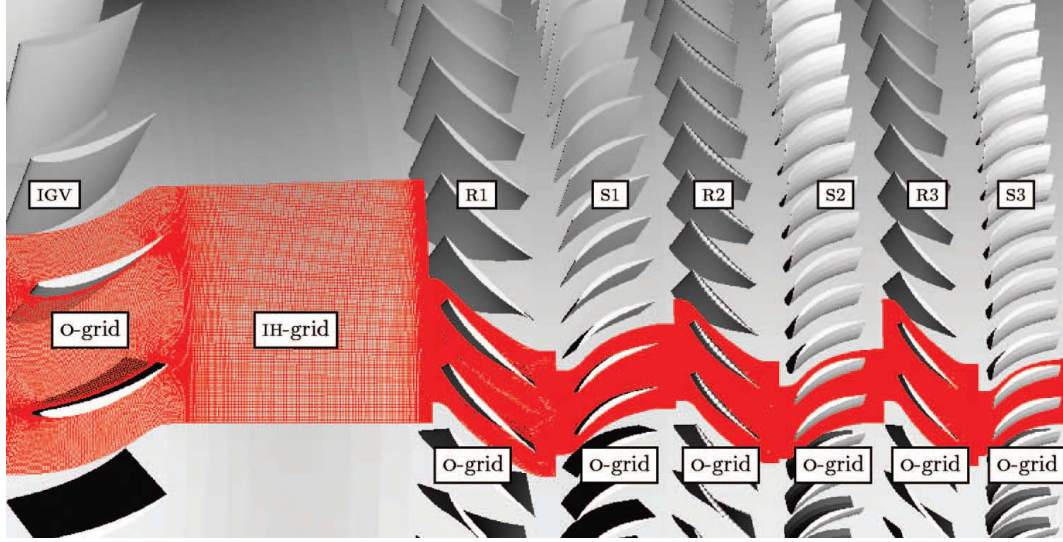


Figure 2.9: Compressor mesh used for steady state compressor simulations by [37].

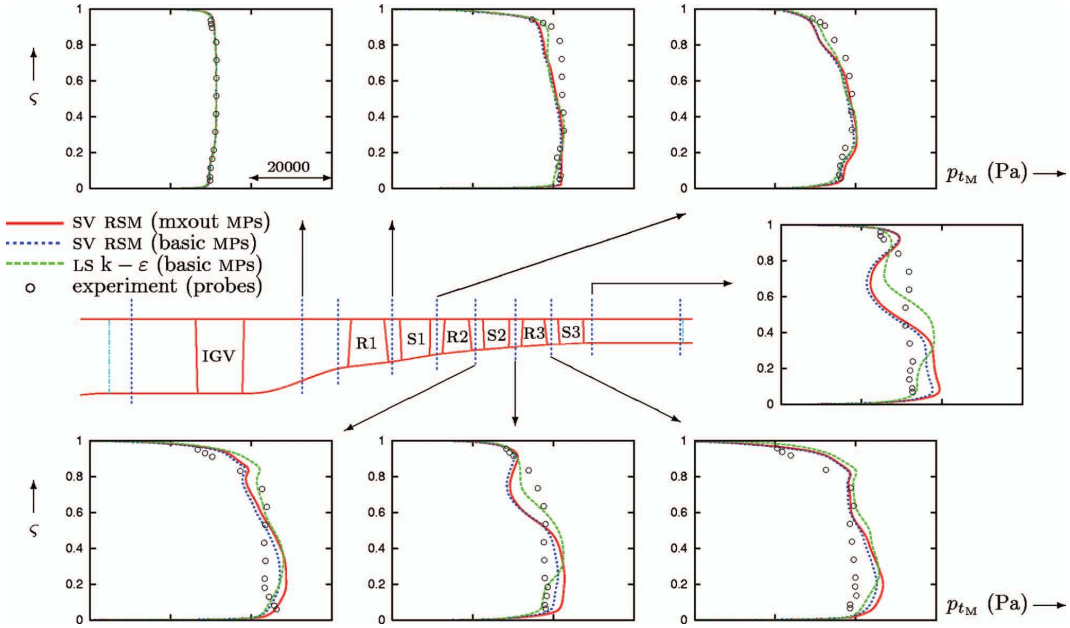


Figure 2.10: Spanwise averaged total pressure plots at various machine locations as shown by [37].

Laumert et al. [38] conducted an experimental and numerically based study of the VKI BRITE EURAM test turbine stage. Authors used both unsteady and steady CFD, the steady CFD analysis which used mixing planes for the stage interface showed poor prediction on the front suction side of the rotor blade where the static pressures were highly overestimated. However the analysis was able to predict the weak shock location at the rear suction side. At a rotational velocity of 6000 rpm the steady CFD showed good agreement with the measurements. Moreover the authors showed no prediction improvement when a low Reynolds number turbulence mode was used without wall functions. Giangiacomo et al. [39] used a steady CFD analysis using mixing plane interface between the blade rows to model a transonic axial stage as well. The authors proposed a best practice on how to balance flow rates between stator and rotor blade rows and stator-rotor pressure interface modeling. The authors showed a linear dependence on the mass flow rate with the guessed hub static pressure. The interface model was shown to create a filtering effect, and 'washes' out the stator wake and trailing edge shock. Ingram and Gregory Smith in [40] detail the discovery of a novel secondary flow feature observed in turbine blading. The feature was discovered by the authors whilst trying to observe the aerodynamics of a non-axisymmetric end wall contouring. The feature allows authors to delineate part of the inlet boundary layer on the blade row endwall as it is being over-turned and rolled up into the passage vortex. Though endwall contouring can provide loss reduction capabilities the sophisticated 3D flows can give rise to adverse features that CFD in most instances fails to predict.

Rosic et al [41] explain the importance of modeling the seal leakage effects in multistage calculations. A multistage calculation is conducted and compared to experimental results, authors postulate the importance of seal leakage modeling to provide a better model of the physics, however, seal leakage modeling is hard to perform as a precise value of seal clearances/leakage values is hard to measure or predict at operating conditions. Muth et al.[42] used various CFD codes to compare numerical versus experimental predictions for Low Reynolds number turbine blade designs. The codes used were the academic code TRACE

and the commercial code ANSYS CFX. Both codes incorporated a transition model based the usage of two transport equations for intermittency and momentum thickness Reynolds numbers. The authors found both codes predicted an early reattachment and therefore a smaller separation bubble.

2.6.2 Unsteady Modeling of Axial Turbines

A steady-state analysis of turbomachinery is inherently deficient as it averages the time dependent interactions that occur between blade rows. Barter et al. [43] used a time-accurate multistage Navier-Stokes solver to model various stator-rotor interaction effects in a transonic turbine. The authors used two approaches, the first technique imposed a specified time-dependent distortion from the adjacent row, and the second technique used a fully coupled stage analysis. The imposed distortion failed to capture the waves that are reflected back and forth from the blade rows, whereas the second method managed to capture all the physical phenomena well when compared to corresponding experimental data. The same solver was used by Van Zante et al. [44] to model a highly loaded axial compressor. The compressor was modeled using a single passage, multiple passages and a half annulus. The authors compared in this case a fully coupled stage analysis for the half annulus versus phase-lag boundary conditions for the single and multiple passage models. The authors in [44] note faster convergence for the half annulus case versus the multiple passage case with phase-lag boundary conditions and showed large circumferential variations in thermodynamic properties as observed in the half-annulus case. Lastly, a maximum of 1.0 point adiabatic efficiency difference was observed between the single passage case and the half annulus simulation.

Pullman [45] uses a transient URANS solver to analyze stator-rotor interactions in a single turbine stage. Authors in [45] note that turbine wake are first compressed as they impinge near the leading edge of downstream blade, and then are bowed and stretched as the wakes convect through blade passage. On the other hand for a compressor, the stretching of the wake within the downstream blade row reduces the velocity deficit so the loss incurred by mixing of the wake within the row is less than if it had mixed out

upstream of the row. Pullan [45] showed total loss generated in a steady calculation is less than the loss generated by the unsteady simulation, and the additional loss generated in the row for the unsteady case is caused by three factors, mainly: mixing out of the inlet flow, unsteady interaction of the inlet flow with the blade row boundary layers and finally mixing associated with the rotor secondary flows over and above that of the rotor flows in the steady calculations. Figure 2.11 shows the numerical mesh used by Pullan [45] for unsteady wake interaction analysis. Figure 2.12 and Figure 2.13 show contour plots of the time averaged rotor exit absolute yaw angle and the stator exit total pressure respectively as calculated by [45].

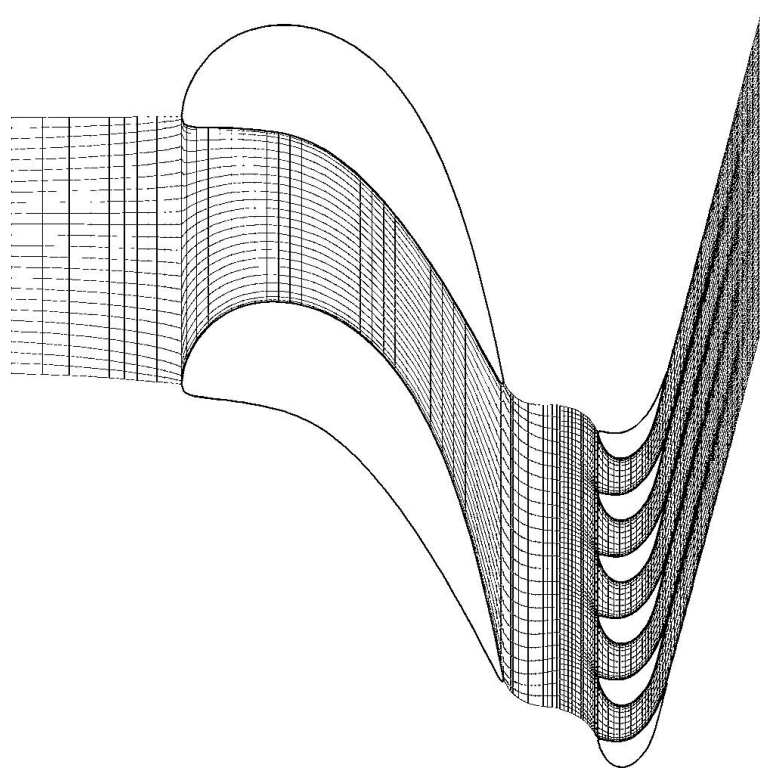


Figure 2.11: Turbine unsteady simulation mesh used by Pullan in [45].

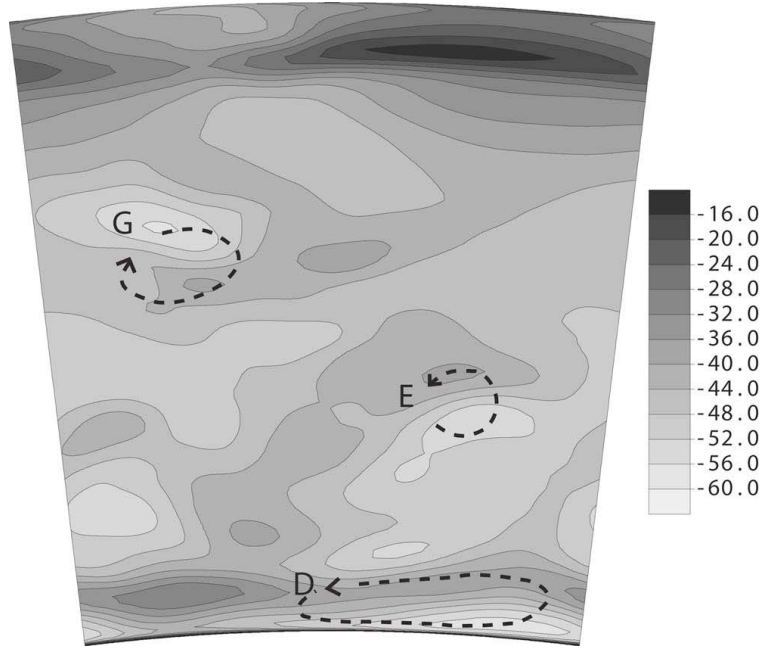


Figure 2.12: Time averaged rotor exit yaw angles as calculated by Pullan in [45].

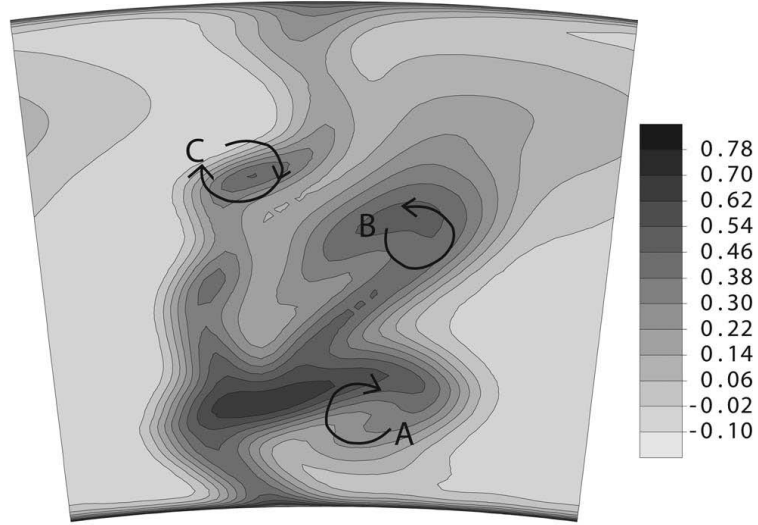


Figure 2.13: Predicted total pressure contour at stator exit as calculated by Pullan in [45].

In another paper by Casciaro et al. [46], the authors used the CFX-TascFlow code to model a pin/blade configuration to observed time periodic interactions between the pin

and blade setup. The authors in a detailed analysis show that each row creates an unclean environment for the preceding and following rows, and unsteady interactions are generally convected downstream because of small axial distances between rows which does not allow time for any mixing out of wakes and flow features exiting a preceding row. Casciaro et al. [46] uses the negative jet theory which was initially brought forth by Meyer [47] to explain unsteady behavior in compressor stages. Casciaro et al. [46] note that some steady state simulations in literature could be accurate within 10% but the secondary flow distribution would be totally wrong, the authors note that a wake is best defined through the velocity field, and the wake does not anchor at the leading edge, but rather the absolute total temperature and the pressure fields associated with the wake do anchor. When the wake hits the leading edge, two zones are created with low unsteadiness, these are termed quiescent zones, and the unsteady transport mechanisms appear to be guided by the creation and downstream movement of the quiescent zones, or they are dependent upon the feeding and discharging mechanisms of the negative jet.

Another study of mixing of stator wakes in turbomachinery was conducted by Rose and Harvey [48] in an effort to understand wake mixing loss in steady flow when the wake has total temperature deficit as well as total pressure deficits, as well as the interaction of stator wakes with downstream rotors and its effect on the mixing loss. Rose and Harvey [48] find the combined total temperature and total pressure wakes tend to have less loss than the total temperature effect alone would give. In a turbine authors discover that less work is extracted if the wake is of low pressure or low total temperature, correspondingly for a compressor more work is done on the wake if it is of low total pressure both these observations result in a reduction of wake mixing losses. Authors also state in a high pressure turbine it is not enough to consider the effects of total pressure wakes alone as the mixing loss may be much higher, and the sensitivity of this loss to acceleration or diffusion would be different.

Denos et al. [49] study the unsteady field pressure measured around the rotor mid-span profile of a VKI turbine test facility. The flow field is modeled with a $k-\varepsilon$ turbulence

model and a modification of the Abu-Ghannam and Shaw [50] correlation for the onset of transition, and the length of transition is modeled using Michelassi et al. [51]. The rotor-stator interaction is driven by upstream wakes, which greatly influence the boundary layer formed on the subsequent downstream blade row. As per Halstead et al. [52] the calming effect was experimentally shown to suppress flow separation and would allow designers to increase loading in the aft region. Halstead et al. [52], highlight the importance of including boundary layer transition effects when using predictive Navier-Stokes solvers.

Chakka and Schobeiri [53] provide a novel transition model that fully incorporates the effects of wake-induced turbulence on boundary layer transition from laminar to turbulent flow. The study in [49] also focuses on interpretation of the unsteady aerodynamics around the rotor mid-section using both experimental results and numerical predictions generated using unsteady CFD. Experimental results were taken using Kulite sensors placed on 3 rotor blades, and data was phase locked averaged, whilst taken into account the slow down of the rotor due to the use of a blow-down facility. Authors observe a sharp pressure rise due to shock, the shock is observed to be unsteady and oscillates slightly around the axial direction. The authors show the comparison between calculated and measured pressures, which compares very well, as the computations reproduce the main features of the flow. The trailing edge shock projects a strong static pressure discontinuity as predicted. The unsteady blade force is dominated by the shock sweeping phenomena, and at design conditions the fluctuation of the axial and tangential forces amounts to 29 and 11% of their mean value respectively. Haldeman et al. [7] ran a numerical and experimental study on a vaneless counter-rotating turbine to study unsteady airfoil interactions and compare CFD predicted data with the experimentally measured phenomena. Again Kulite pressure transducers were used to measure unsteady pressure. The experiment is designed to simulate the flow function at the desired inlet total pressure and total temperature. A slip ring was used to connect the Kulite transducers in the rotating rows. On the numerical side, a multi-sector Navier-Stokes code that utilizes a mixing length (Baldwin-Lomax) turbulence model was used to model the counter-rotating turbine. Authors conclude that the code is

conservative in its prediction of the pressure interaction between the counter-rotating blade rows. Moreover, in Clark et al. [54], the effect of scaling the airfoil on the unsteady CFD results is investigated. The airfoil scaling is done to save on computational requirements needed by the unsteady model. The authors show that without scaling, the code at full mesh was able to predict the level of unsteadiness of the blade well, however with scaling the results were shown to not match the experimental data, and as a result the authors warn from using scaled results to reduce high levels of unsteadiness at multiples of the relevant passing frequency. Bohn et al. [55] use a 3D unsteady code to analyze the influence of blade design on the blade row interaction. A comparison is made between experimental and unsteady CFD data for a bowed blade design. The bowed blade is designed to reduce secondary flow effects or losses at the tip and casing regions. The numerical results showed good matching with the corresponding experimental results obtained from the Aachen University turbine rig.

Gehrer et al. [56] show that loss predicted by unsteady CFD showed much higher loss values than that showed by the steady or time averaged loss in a transonic stage. Moreover, the effect of unsteadiness causes higher losses as indicated by an increase in friction and also effects the turning of the flow resulting in an increase in the exit flow by 1 degree. The numerical results overall showed good agreement with Laser-Doppler Velocimetry (LDV) at similar Strouhal numbers, however the unsteady CFD showed a higher velocity defect further downstream in the near wake region. A similar bowed versus straight stator blades study was conducted by Vand and Songtao [57], where the authors observed stronger rotor trailing wakes near the hub and casing region of the blade. In positively bowed blades, the starting point of the horseshoe vortex branch point is shifted towards the upstream, and the flow passage vortex occurs sooner. Although the passage vortex initiates earlier its strength and scale is reduced. The bowed blades also showed reduction in loss coefficients at the endwalls versus the straight stacked blades. In addition to the bowing study versus studies have been conducted testing the effect of clearances on shrouded and unshrouded turbine by Yoon et al. [58]. The authors looked at two levels of reaction, and observed the

tip clearance flow interaction with the main throughflow, and at the lower reaction operation there is less of a tip leakage penalty than at the higher 50% reaction. El-Ghandour et al. [59] used a Detached Eddy Simulation (DES) solver to analyze tip leakage flows over a stationary GE- E^3 blade, showing the vortex formation depends on vorticity imparted with the incoming flow, the flow with higher velocity gradients caused larger vortices. This was found to be valid for both cavity vortices as well as the leakage vortex.

Turbulence modeling has achieved significant strides in the past 2 decades and the mostly widely used models in turbomachinery applications include the two-equation based $k - \varepsilon$ and $k - \omega$ models, simple as they are when compared to more advanced models such as the 7 equation Reynolds Stress models or large eddy simulation (LES) models. Nonetheless, $k - \varepsilon$, $k - \omega$ and the hybrid SST model introduced by [60, 61] still provide a robust and accurate platform for turbomachinery based CFD applications. Mumic et al. [62] presented a comparison between the performance of three different turbulence models in modeling the heat transfer and fluid flow in a single 3D passage. Heat transfer and fluid flow results were compared with experimentally obtained values. The different models showed good agreement with experimental data. Sanz et al. [63] modeled transonic flow through a pre-swirl turbine inlet guide vane using two different numerical solvers and compared their numerical data with total pressure measurements in the downstream plane, with both models showing good prediction of the overall loss scheme. Mischo et al. [64] used CFD analysis on a single rotor to quantify the performance of different blade tip schemes. Different recess cavities were modeled and compared to various experimental results and a novel recess geometry was introduced based on the optimization study performed using CFD. A similar optimization study was performed by Adami et al. [65] to observe the effect of different shroud configurations using unsteady CFD. Two different shroud configurations were used to analyze the cavity flow and its influence on the mainstream flow vortices. Moreover, a study conducted by Medic et al. [66] a coupled solver that uses URANS methodology in the turbomachinery components and a Large Eddy Simulation LES code in the combustor region. A novel approach is presented which enables

the coupling of both codes, where the coupling is based on the fact that in full operating conditions the mean flow at the interfaces is non-uniform and local turbulence production dominates convection effects. The interfacing between RANS and LES is done using a duct RANS quasi-2D code or solution using the input LES solution at the interface, which provide the turbulence parameters for the RANS code, the mean values are passed to the URANS solver from the LES solution.

3. TRANSITIONAL FLOW REGIMES

Boundary layer transition is a process by which the boundary layer gradually develops from a laminar state to a turbulent state. The transition process under a steady flow field can occur via three types of transition process:

- Natural Transition
- Bypass Transition
- Separated Flow Transition

3.1 Natural Transition

Transition was first investigated by Osborne Reynolds pipe flow experimentation [67], whereby a dye was injected into a pipe flow, through observation Reynolds was able to establish a criteria for transition from smooth laminar flow to the more chaotic turbulent flow regimes. This transition criteria was found to occur at a Reynolds number of ≈ 2300 for pipe flow. An investigation into the transition of a plate flow was first investigated independently by Burgers [68] and van der Hegge [69] who found that the laminar boundary layer develop linear oscillations of well defined frequency when the Reynolds number exceeds a specific value. To explain this phenomena several authors used various stability and perturbation methods, such techniques were utilized by Rayleigh [70], Prandtl [71] and Schlichting [72]. Through the stability analysis methods a well known stability equation was derived, and termed the Orr-Sommerfeld equation. From the eigenvalues of this equation the stability loci can be mapped for the flow under consideration. Tollmien [73] was first to apply the stability equation for the Blasius profile for laminar external boundary layer flow over a plate, in process finding the stability points for laminar flow over a flat plate. The experimentally noted disturbances that occur at a critical point were therefore called Tollmein-Schlichting waves which appear in the transitional flow between laminar and turbulent flow over a flat plate.

3.2 Separated Flow Transition

Separated flow transition occurs in the shear layer, after a flow separates it reattaches as a turbulent flow and is called a laminar separation and turbulent reattachment bubble. Figure 3.1 illustrates the reattachment bubble phenomena. Mayle [74] provides a thorough review of the various experimental results that investigate separated flow transition.

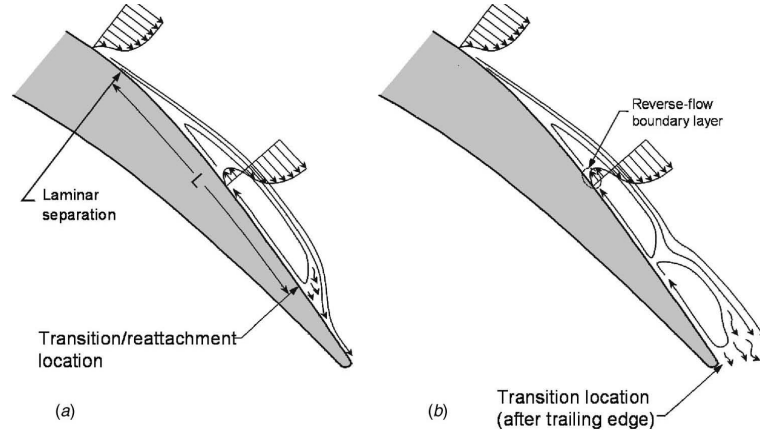


Figure 3.1: Representation of suction-side, laminar separation showing both reattached (a) and stalled (b), obtained from [75].

3.3 Bypass Transition

Emmons [76] was first to model the transition via a novel approach by assuming every point in the boundary layer as turbulent for at least a fraction of time. The disturbances in the transition region were considered as turbulent spots and developed expressions for the generation and the development of these spots in the XY plane as a function of time. The turbulent spot theory was later developed by Dhawan and Narashima [77] who developed expressions for the intermittency factor in the transition region. Morkovin [78] discovered that the conventional Orr-Sommerfeld equation was not applicable for turbulence intensities as high as 10%, and that at such high turbulence intensities the Tollmien-Schlichting waves were bypassed and the turbulence spot formation occurred right away. Gostelow

and Blunden [79] also reported non-linear bypass effects through transition that are independent of Tollmien-Schlichting waves. Therefore, under high turbulence intensity the transition process is termed bypass transition.

3.4 Transition in Turbomachinery

Since flow in turbomachinery is periodically unsteady, the transition process is therefore, as a result, seen as a transient or unsteady process. This unsteadiness is due to the unsteady wakes from the upstream rotor and is called wake induced transition as termed by Mayle [74]. Wake induced transition is seen to bypass the natural transition process, however the process of wake induced transition is not fully understood. Reverse transition is another phenomena encountered in turbomachinery, whereby a transition from turbulent to laminar flow can occur at low turbulence levels when an acceleration parameters exceeds a specific value. This kind of transition is also discussed by Mayle [74] and is of great importance to the turbine designer.

Overall, transition is a complex phenomena that is affected by numerous parameters which include free-stream turbulence, curvature, pressure gradient. Various models have been introduced that attempt to accurately model the transition process. Correctly accounting for the transition can provide for better compressor and turbine losses predictions especially in the case of low pressure turbines. Some of the most common models that have been used include the Abu Ghannam and Shaw [50] and the models developed by Menter [80, 81]. For the Menter based models two transport equations are solved for the intermittency and the transition momentum thickness based Reynolds number (greater details available in [80], but equations are listed here for reference):

$$\frac{\partial(\rho\gamma)}{\partial t} + \frac{\partial(\rho U_i \gamma)}{\partial x_j} = P_{\gamma 1} - E_{\gamma 1} + P_{\gamma 2} - E_{\gamma 2} + \frac{\partial}{\partial x_j} \left[\left(\mu + \frac{\mu_t}{\sigma_f} \right) \frac{\partial \gamma}{\partial x_j} \right] \quad (3.1)$$

$$\frac{\partial(\rho \widetilde{Re_{\theta t}})}{dt} + \frac{\partial(\rho U_j \widetilde{Re_{\theta t}})}{\partial x_j} = P_{\theta t} + \frac{\partial}{\partial x_j} \left[\sigma_{\theta t} (\mu + \mu_t) \frac{\partial \widetilde{Re_{\theta t}}}{\partial x_j} \right] \quad (3.2)$$

The transition model of Menter [80] interacts with the Shear Stress Transport turbulence (SST) model via:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho U_j k)}{\partial x_j} = \widetilde{P}_k - \widetilde{D}_k + \frac{\partial}{\partial x_j} \left[(\mu + \sigma_k \mu_t) \frac{\partial k}{\partial x_j} \right] \quad (3.3)$$

where the production term \widetilde{P}_k is modified (only in the k-equation) as:

$$\widetilde{P}_k = \gamma P_k \quad (3.4)$$

The original production term of the SST model [80] was defined as:

$$P_k = \min \left(\tau_{ij} \frac{\partial U_i}{\partial x_j} \right) \quad (3.5)$$

And the destruction term is modified as:

$$\widetilde{D}_k = \min (\max (\gamma_{eff}, 0.1) 1.0) D_k \quad (3.6)$$

The original destruction term of the SST model [80] was defined as:

$$D_k = \frac{9}{100} k \omega \quad (3.7)$$

The various definition for the source and destruction terms can be found in [80], this model has been tested with various cases in [81] and showed very good correspondence with experimental data.

Chakka and Schobeiri [53] developed a transition model geared towards curved plates and linear turbine blade cascades, which describe boundary layer transition as a function of upstream wake passing period or frequency. The model prescribes various intermittency values as a function of transition start and end Reynolds numbers. The transition start and end Reynolds numbers are obtained in turn as a function of the upstream wake passing frequency. The model was incorporated into the boundary layer code TEXSTAN (a

modification of STAN5) and the measured heat transfer coefficient was compared to the calculated values showing good agreement. Gier et al. [82] analyzed a three stage LPT using a finite volume solver that incorporated a two-equation turbulence model and a version of the Abu-Ghannam and Shaw (AGS) [50] transition model. The results were compared to experimental results obtained from a closed circuit gas rig with adjustable inlet and exit conditions to simulate different turbine operating conditions. The code uses a multistage interface model or mixing plane interfaces with non-reflecting boundary interfaces. The numerical solution was shown to predict separation and reattachment points pretty well when compared to the experimental observations. However, in some cases the transition model was shown to predict transition further downstream than the experimental observations have shown. The pattern was mainly seen at regions adjacent to the blade endwalls. Hu and Fransson [83] observed the AGS model to perform best when subjected to weak pressure gradients, but still predicts a later point of transition under low Reynolds number conditions, whilst failing to capture the transition which is dominated by strong and adverse pressure gradients. Moreover AGS was shown to perform best at turbulence intensity values that are greater 6%. Babajee and Arts [84] use the Menter transition correlation, implemented in the VKI code elsA on two LPT test cases, with the same compressible Zweifel loading coefficient but at different blade loading configurations. The numerical results were shown to give good results when compared to experimental testing conducted at VKI. The transition onset, transition end and separation show good agreement when compared to experimental flow field observations showing promising capabilities in transition onset modeling and predictions.

4. RESEARCH OBJECTIVES

The use of computational fluid dynamics in design and analysis of turbomachinery has become a critical activity with the greater usage of CFD in the multiple stages of turbomachinery design. The confidence level of CFD results is dependent of various factors including gridding, boundary condition selection, mixing planes, turbulence modeling and transition modeling among others, this study aims to look at various blade design that have been analyzed using CFD and tested experimentally and to study the trends exhibited between predicted performance and flow visualization versus measured performance and flow visualization. Both steady-state CFD or RANS, as well as unsteady CFD or URANS are used, and compared to data. Additionally a further and more detailed look at transition modeling is taken, by using simple examples such as a stationary blade cascade as well as a curved plate flow to look at the state of the art with regards to transition modeling and what is the path ahead towards achieve accurate and reliable laminar-to-turbulent transition. The specific research objectives can be summarized as:

4.1 Numerical Mesh

- Create high quality structured mesh for 2-Stage Turbine
 - Ensure mesh quality at wall locations and minimum y^+ values at near wall locations
 - Ensure mesh convergence via mesh sensitivity analysis
- Create high quality structured mesh for 3D bowed 3-Stage Turbine
 - Ensure mesh quality at wall locations and minimum y^+ values at near wall locations
 - Ensure mesh convergence via mesh sensitivity analysis
- Create high quality structured mesh for cylindrical 3-Stage Turbine

- Ensure mesh quality at wall locations and minimum y^+ values at near wall locations
- Ensure mesh convergence via mesh sensitivity analysis

4.2 Steady State Analysis

- Perform steady state analysis for 3 cases: 2-stage turbine design, cylindrical 3-stage turbine and 3D bowed 3-stage turbine
- Evaluate performance of interface modeling
- Evaluate performance at various machine conditions, off-design and on-design
- Compare data predicted with data measured from experimentation
- Compare 3D bowed blading to 2D straight blade flow characteristics
- Compare machine overall efficiency predicted to machine overall efficiency measured

4.3 Transient Analysis

- Perform time accurate transient analysis for 3 cases: 2-stage design, cylindrical 3-stage design and 3D bowed 3-stage turbine design
- Evaluate performance of averaged transient versus steady state analysis when compared to experimental data
- Evaluate overall machine efficiency using transient analysis versus steady state analysis and experimental data
- Evaluate overall performance of Shear Stress Transport Equation [60, 61]
- Suggest future paths for more accurate and reliable CFD a-priori prediction of turbomachinery designs.

4.4 Transition Modeling in Turbomachinery and Boundary Layer Flows

- Implement transition model by Chakka and Schobeiri [53] into steady RANS Code for curved plate and stator cascades
- Evaluate performance of transition model [53] when compared to Experimental data
- Discuss future direction with regards to the transition modeling with an emphasis on turbomachinery

5. COMPUTATIONAL FLUID DYNAMICS AND TURBULENCE MODELING

In this chapter a brief overview of computational fluid dynamics and the finite volume method will be discussed as well as turbulence modeling in turbomachinery applications. Computational fluid dynamics is a branch of fluid mechanics where numerical methods are used to algebraically represent the Navier-Stokes differential equations. The algebraic schemes are then used to provide solutions to the equations which represent physical fluid flow fields. The field's advancement has been largely dependent on the advancement of computational platforms in the last thirty years that have enabled more complex and expensive solutions. Increased memory has enabled more accurate solutions with less abstract models or approximation to the differential equations or boundary conditions. The general equations of motion for incompressible flows with constant transport properties include the continuity equation

$$\nabla \cdot \vec{V} = 0 \quad (5.1)$$

The Navier stokes equations:

$$\frac{D\vec{V}}{Dt} = -\frac{1}{\rho}\nabla p + \vec{g} + \nu\Delta\vec{V} \quad (5.2)$$

The energy equation:

$$\rho c_p \frac{DT}{Dt} = k\nabla^2 T + \Phi \quad (5.3)$$

where the dissipation function Φ is expressed as:

$$\Phi = \mu \left(\frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i} \right) \frac{\partial V_i}{\partial x_j} \quad (5.4)$$

For incompressible flow conditions the continuity equations may be simplified to:

$$\frac{\partial V_1}{\partial x_1} + \frac{\partial V_2}{\partial x_2} + \frac{\partial V_3}{\partial x_3} = 0 \quad (5.5)$$

The shear stress tensor τ_{ij} can be expanded to:

$$\tau_{ij} = \begin{bmatrix} \tau_{11} & \tau_{12} & \tau_{13} \\ \tau_{21} & \tau_{22} & \tau_{23} \\ \tau_{31} & \tau_{32} & \tau_{33} \end{bmatrix} \quad (5.6)$$

For an incompressible system where $\nabla \cdot \vec{V} = 0$ the stress tensor can be written as:

$$\tau_{ij} = \begin{bmatrix} 2\mu \frac{\partial V_1}{\partial x_1} & \mu \left[\frac{\partial V_2}{\partial x_1} + \frac{\partial V_1}{\partial x_2} \right] & \mu \left[\frac{\partial V_1}{\partial x_3} + \frac{\partial V_3}{\partial x_1} \right] \\ \mu \left[\frac{\partial V_2}{\partial x_2} + \frac{\partial V_1}{\partial x_2} \right] & 2\mu \frac{\partial V_2}{\partial x_2} & \mu \left[\frac{\partial V_3}{\partial x_2} + \frac{\partial V_2}{\partial x_3} \right] \\ \mu \left[\frac{\partial V_1}{\partial x_3} + \frac{\partial V_3}{\partial x_1} \right] & \mu \left[\frac{\partial V_3}{\partial x_2} + \frac{\partial V_1}{\partial x_3} \right] & 2\mu \frac{\partial V_3}{\partial x_3} \end{bmatrix} \quad (5.7)$$

The momentum equations for incompressible flow may be expanded as:

$$\rho \left(\frac{\partial V_1}{\partial t} + V_1 \frac{\partial V_1}{\partial x_1} + V_2 \frac{\partial V_1}{\partial x_2} + V_3 \frac{\partial V_1}{\partial x_3} \right) = -\frac{\partial p}{\partial x_1} + \mu \left(\frac{\partial^2 V_1}{\partial x_1^2} + \frac{\partial^2 V_1}{\partial x_2^2} + \frac{\partial^2 V_1}{\partial x_3^2} \right) + \rho g_1 \quad (5.8)$$

$$\rho \left(\frac{\partial V_2}{\partial t} + V_1 \frac{\partial V_2}{\partial x_1} + V_2 \frac{\partial V_2}{\partial x_2} + V_3 \frac{\partial V_2}{\partial x_3} \right) = -\frac{\partial p}{\partial x_2} + \mu \left(\frac{\partial^2 V_2}{\partial x_1^2} + \frac{\partial^2 V_2}{\partial x_2^2} + \frac{\partial^2 V_2}{\partial x_3^2} \right) + \rho g_2 \quad (5.9)$$

$$\rho \left(\frac{\partial V_3}{\partial t} + V_1 \frac{\partial V_3}{\partial x_1} + V_2 \frac{\partial V_3}{\partial x_2} + V_3 \frac{\partial V_3}{\partial x_3} \right) = -\frac{\partial p}{\partial x_3} + \mu \left(\frac{\partial^2 V_3}{\partial x_1^2} + \frac{\partial^2 V_3}{\partial x_2^2} + \frac{\partial^2 V_3}{\partial x_3^2} \right) + \rho g_3 \quad (5.10)$$

The energy equation for incompressible flow may be expanded as per [85]:

$$\begin{aligned} c_p \left(\frac{\partial T}{\partial t} + V_1 \frac{\partial T}{\partial x_1} + V_2 \frac{\partial T}{\partial x_2} + V_3 \frac{\partial T}{\partial x_3} \right) &= \frac{k}{\rho} \left(\frac{\partial^2 T}{\partial x_1^2} + \frac{\partial^2 T}{\partial x_2^2} + \frac{\partial^2 T}{\partial x_3^2} \right) \\ &+ \frac{1}{\rho} \left(\frac{\partial p}{\partial t} + V_1 \frac{\partial p}{\partial x_1} + V_2 \frac{\partial p}{\partial x_2} + V_3 \frac{\partial p}{\partial x_3} \right) + \frac{1}{\rho} \phi \end{aligned} \quad (5.11)$$

The equations shown above are a complex system of nonlinear partial differential equations, and are difficult to solve analytically. The term Navier-Stokes can sometimes be attributed in literature to the entire system of equations which includes, continuity, momentum and energy equations. All the equations listed above, in conservation form may

be expressed as:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x_1} + \frac{\partial G}{\partial x_2} + \frac{\partial H}{\partial x_3} = J \quad (5.12)$$

Equation (5.12) is used to represent the entire system of governing equations:

$$U = \left\{ \begin{array}{c} \rho \\ \rho V_1 \\ \rho V_2 \\ \rho V_2 \\ \rho \left(e + \frac{V^2}{2} \right) \end{array} \right\} \quad (5.13)$$

$$F = \left\{ \begin{array}{c} \rho V_1 \\ \rho V_1^2 + p - \tau_{11} \\ \rho V_2 V_1 - \tau_{12} \\ \rho V_3 V_1 - \tau_{13} \\ \rho \left(e + \frac{V^2}{2} \right) V_1 + p V_1 - k \frac{\partial T}{\partial x_1} - u \tau_{11} - V_2 \tau_{12} - V_3 \tau_{13} \end{array} \right\} \quad (5.14)$$

$$G = \left\{ \begin{array}{c} \rho V_2 \\ \rho V_1 V_2 - \tau_{21} \\ \rho V_2^2 + p - \tau_{22} \\ \rho V_3 V_2 - \tau_{23} \\ \rho \left(e + \frac{V^2}{2} \right) V_2 + p V_2 - k \frac{\partial T}{\partial x_2} - V_1 \tau_{21} - V_2 \tau_{22} - V_3 \tau_{23} \end{array} \right\} \quad (5.15)$$

$$H = \left\{ \begin{array}{c} \rho V_3 \\ \rho V_1 V_3 - \tau_{31} \\ \rho V_2 V_3 - \tau_{32} \\ \rho V_3^2 + p - \tau_{33} \\ \rho \left(e + \frac{V^2}{2} \right) V_3 + p V_3 - k \frac{\partial T}{\partial x_3} - u \tau_{31} - V_2 \tau_{32} - V_3 \tau_{33} \end{array} \right\} \quad (5.16)$$

$$J = \begin{pmatrix} 0 \\ \rho f_1 \\ \rho f_2 \\ \rho f_3 \\ \rho(V_1 f_1 + V_2 f_2 + V_3 f_3) + \rho \dot{q} \end{pmatrix} \quad (5.17)$$

The column vectors F, G , and H are termed the flux terms, and J represents a source term. U would provide a solution vector to this system. The system is marched in time and a solution calculated at each time step and to facilitate this representation equation (5.12) can be re-arranged to:

$$\frac{\partial U}{\partial t} = J - \frac{\partial F}{\partial x_1} - \frac{\partial G}{\partial x_2} - \frac{\partial H}{\partial x_3} \quad (5.18)$$

5.1 Reynolds Averaged Navier Stokes (RANS)

In order to discuss the Reynolds Averaged Navier Stokes equations one needs to address turbulence. Reynolds number of a flow describes a ratio of inertial forces to viscous forces. Experiments conducted first by Osborne Reynolds observed that at values below a specific Reynolds number, called the critical Reynolds number, the flow is seen to be smooth, if there are no time dependent effects or boundary conditions the flow is observed to be steady. This flow is termed to laminar. At or above the critical Reynolds number the flow is seen to be random in nature, with unsteady characteristics. This is called turbulent flow, the turbulent flow can be decomposed into a steady mean value \bar{V} , with a fluctuating component $V'(t)$. For a statistically steady flow, the overall flow can be decomposed into a turbulent component and an averaged component $V(t) = \bar{V} + V'(t)$. Therefore in a 3D flow one will observe the averaged values of the three velocity components, $\bar{V}_1, \bar{V}_2, \bar{V}_3$ and pressure \bar{p} and their respective fluctuating components V'_1, V'_2, V'_3 and p' .

The mean Φ of a flow property ϕ can be defined as:

$$\Phi = \frac{1}{\delta t} \int_0^{\delta t} \phi(t) dt \quad (5.19)$$

ϕ is time dependent and can be separated into a mean component Φ and a time-varying fluctuating component ϕ' with zero mean value, therefore:

$$\phi(t) = \Phi + \phi(t)' \quad (5.20)$$

For a statistically steady flow, the time average of the fluctuating component ϕ' is by definition equal to zero:

$$\bar{\phi}' = \frac{1}{\delta t} \int_0^{\delta t} \phi'(t) dt = 0 \quad (5.21)$$

A root mean square evaluation of ϕ to provide information about the fluctuation component of the flow, the root mean square or rms is defined as:

$$\phi_{rms} = \sqrt{(\bar{\phi}')^2} = \left[\frac{1}{\delta t} \int_0^{\delta t} (\phi')^2 dt \right] \quad (5.22)$$

The kinetic energy associated with the turbulent flow is defined as:

$$k = \frac{1}{2} \left(\overline{V_1'^2} + \overline{V_2'^2} + \overline{V_3'^2} \right) \quad (5.23)$$

Turbulence intensity Tu is related to kinetic energy and reference mean flow and is defined as:

$$Tu = \frac{\sqrt{\frac{2}{3}k}}{U} \quad (5.24)$$

As clearly outlined in Schobeiri [85], a turbulent quantity like velocity is decomposed in a mean and fluctuating component. The following can be obtained from the averaging process:

$$\begin{aligned} \frac{\partial \bar{V}_i}{\partial x_i} &= \frac{\partial \bar{V}_i}{\partial x_i} \\ \bar{V}_i &= \overline{\bar{V}_i + V_i'} = \bar{V}_i \end{aligned}$$

since, averaging the fluctuating component $\overline{V'_i} = 0$.

$$\overline{\overline{V_i V_i}} = \overline{V_i V_i}$$

$$\overline{\overline{V_i V'_i}} = \overline{V_i V'_i} = 0$$

$$\overline{\overline{V'_i V'_i}} = \overline{V'_i V'_i}$$

$$\overline{\overline{V'_i V'_j}} = \overline{V'_i V'_j}$$

Substituting $V_i(t) = \overline{V_i} + V'_i(t)$, and $p(t) = \overline{p} + p'(t)$ into the Navier-Stokes equations and averaging gives the averaged continuity equation:

$$\frac{\partial V_i}{\partial x_i} = 0 \quad (5.25)$$

after averaging results in

$$\frac{\partial \overline{V_i}}{\partial x_i} \quad (5.26)$$

For the momentum equations:

$$\frac{\partial V_i}{\partial t} + V_j \frac{\partial V_i}{\partial x_j} = g_i - \frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 V_i}{\partial x_j \partial x_j} \quad (5.27)$$

resulting in:

$$\frac{\partial \overline{V_i}}{\partial t} + \overline{V_j} \frac{\partial \overline{V_i}}{\partial x_j} = \overline{g_i} - \frac{1}{\rho} \frac{\partial \overline{p}}{\partial x_i} + \nu \frac{\partial^2 \overline{V_i}}{\partial x_j \partial x_j} - \frac{\partial \overline{V'_i V'_j}}{\partial x_j} \quad (5.28)$$

which can be re-arranged to:

$$\rho \frac{\partial \overline{V_i}}{\partial t} + \rho \overline{V_j} \frac{\partial \overline{V_i}}{\partial x_j} = \rho \overline{g_i} + \frac{\partial}{\partial x_j} \left[-\overline{p} \delta_{ij} + 2\mu D_{ij} - \rho \overline{V'_i V'_j} \right] \quad (5.29)$$

where, D_{ij} is the mean rate of strain tensor and is defined as $\bar{D}_{ij} = \frac{1}{2} \left(\frac{\partial \bar{V}_i}{\partial x_j} + \frac{\partial \bar{V}_j}{\partial x_i} \right)$. Integration in time will remove the first term on the left hand side leaving:

$$\rho \bar{V}_j \frac{\partial \bar{V}_i}{\partial x_j} = \rho \bar{g}_i + \frac{\partial}{\partial x_j} \left[-\bar{p} \delta_{ij} + 2\mu D_{ij} - \rho \overline{V'_i V'_j} \right] \quad (5.30)$$

Equation (5.30) is what is commonly known as the Reynolds Averaged Reynolds equations. Note the so-called Reynolds stress term $\rho \overline{V'_i V'_j}$. The extra term or terms creates a closure problem which requires an extra equation to describe the Reynolds stresses. Turbulence modeling is needed to develop computational schemes of sufficient accuracy for users to predict the Reynolds stresses and their transport terms.

5.2 Turbulence Modeling

As stated in the previous section the introduction of an extra term into the RANS equations creates a closure problem due to the extra Reynolds Stress term. There have many different approaches to this problem throughout the past 25 to 30 years, we present a few here:

5.2.1 Zero Equation Models

Most of the earliest turbulence models were based on Prandtl's mixing length hypothesis. Prandtl suggested that the eddy viscosity can be represented by [85]:

$$\mu_t = \rho L_m^2 \left| \frac{\partial V_1}{\partial x_2} \right| \quad (5.31)$$

Inner Eddy Viscosity Model

The mixing length relates the eddy viscosity to the local mean velocity gradient. In order to apply the Prandtl mixing length successfully a method is needed to define the mixing length parameter L_m . The boundary layer can be divided into a inner and outer layer, where the inner layer contains the viscous sublayer, the buffer layer and part of the fully turbulent logarithmic region. The outer layer includes the remaining part of the

logarithmic layer and the wake region. Eddy viscosity in the inner layer follows Prandtl's form and is given by:

$$\mu_{t\,inner} = \rho L_m^2 |\omega| \quad (5.32)$$

where the mixing length is defined by:

$$L_m = \kappa y \left[1 - \exp\left(\frac{-y^+}{A^+}\right) \right] \quad (5.33)$$

Here y is the normal distance to the wall, κ is the von Karaman constant and the vorticity is defined as:

$$\omega = \nabla \times \vec{V} \quad (5.34)$$

Cebeci-Smith Model

Cebeci Smith [86] proposed the outer eddy viscosity may be expressed as:

$$\mu_{t\,outer} = \rho \alpha u_e \delta^* \quad (5.35)$$

The term α is a constant that was assigned a value of 0.0168 for flows where the Reynolds number based on momentum thickness Re_θ is greater than 5000. The term δ^* is the displacement thickness and u_e is the velocity at the edge of the boundary layer. The final eddy viscosity term is:

$$\mu_t = [\mu_{t\,inner}, \mu_{t\,outer}] \quad (5.36)$$

A simple model it requires knowledge of the conditions at the edge of the boundary layer and the boundary layer thickness.

Baldwin-Lomax Model

The Baldwin Lomax model [87] developed an expression for outer eddy viscosity that does not require knowledge of conditions at the edge of the boundary layer. To this day a widely used turbulence model despite its simplicity. The eddy viscosity in the outer layer

is defined as:

$$\mu_{t\ outer} = \rho K C_p F_{wake} F_{kleb} \quad (5.37)$$

where F_{wake} contains the mixing length term and F_{kleb} is the Klebanoff intermittency factor, these factors are defined as:

$$F_{wake} = \min \left\{ y_{max} F_{max}, C_{wk} y_{max} U_{diff}^2 / F_{max} \right\} \quad (5.38)$$

The values F_{max} and y_{max} are taken from the maximum of the F function:

$$F(y) = y |\omega| \left[1 - \exp \left(\frac{-y^+}{A^+} \right) \right] \quad (5.39)$$

and the term U_{diff} is defined by:

$$U_{diff} = \left(\sqrt{V_1^2 + V_2^2} \right)_{max} - \left(\sqrt{U_1^2 + V_2^2} \right)_{min} \quad (5.40)$$

The F function is calculated along a line normal to the wall. The Klebanoff intermittency factor is used to reduce the eddy viscosity to zero at the outer edge of the boundary layer.

And is given by:

$$F_{kleb}(y) = \left[1 + 5.5 \left(\frac{C_{kleb} y}{y_{max}} \right)^6 \right]^{-1} \quad (5.41)$$

The constants used in the Baldwin-Lomax model are: $A^+ = 26$, $C_{cp} = 1.6$, $C_{kleb} = 0.3$, $C_{wk} = 1.0$, $\kappa = 0.4$, and $K = 0.068$

Zero-equation turbulence models or Algebraic models work well for flows that can be characterized by a single length scale like attached boundary layers or simple shear layers. The Baldwin-Lomax model determines the appropriate mixing length from the location of the F function maximum. In more complex flows with multiple shear layers, such as separated boundary layers or wall jets the F function will contain several peaks. When multiple peaks arise it becomes a challenge to determine the proper peak location to use. Various techniques have been presented to facilitate the peak solution procedure.

5.2.2 One-Equation Models

The generalized form of all turbulence models is similar to the form of a species equation in reacting flows:

$$\frac{\partial Z}{\partial t} = +V_i \frac{\partial Z}{\partial x_i} = \frac{1}{\sigma} \frac{\partial}{\partial x_i} \left[\left(\nu + \nu_t \frac{\partial V}{\partial x_i} \right) \right] + P(Z) - D(Z) \quad (5.42)$$

Here Z is a general turbulence variable, and the left hand side of (5.42) represent the convective transport terms. The first term on the right hand side of (5.42) is the diffusion term for parameter Z assuming a gradient diffusion model. The functions $P(Z)$ and $D(Z)$ are the production and destruction terms for Z respectively. The production term for turbulence models are usually functions of the eddy viscosity and/or the vorticity. A functional term between Z and the eddy viscosity is required to complete the model. Examples of the closure needed that have been used in the past 30 years include the Baldwin-Barth model and the Spalart-Allmaras model.

5.2.3 Two Equation Models

A good example of a two-equation turbulence model is the k - ε model, one of the most commonly and implemented two-equation turbulence models. The instantaneous kinetic energy of a turbulent flow is the sum of the mean kinetic energy and the turbulent kinetic energy or:

$$k(t) = K + k'(t) = \frac{1}{2} (V_1^2 + V_2^2 + V_3^2) + \frac{1}{2} (V_1'^2 + V_2'^2 + V_3'^2) \quad (5.43)$$

An equation for the mean kinetic energy K can be obtained by multiplying the various component momentum equations (5.8) to (5.10) by their respective velocity components, a more detailed derivation can be found in [85], however it will be discussed here in brevity. Taking the momentum equation and performing a scalar multiplication with the velocity

vector gives:

$$\begin{aligned} \vec{V}' \cdot \frac{\partial \vec{V}}{\partial t} + \vec{V}' \cdot (\vec{V} \cdot \nabla \vec{V}') + \vec{V} + \vec{V}' \cdot (\vec{V}' \cdot \nabla \vec{V}) + \vec{V}' \cdot \vec{V}' \cdot \nabla \vec{V}' = -\frac{1}{\rho} \vec{V}' \cdot \nabla p' + \\ \nu \vec{V}' \cdot \Delta \vec{V}' - \vec{V}' \cdot \nabla \cdot (\vec{V}' \vec{V}') \end{aligned} \quad (5.44)$$

The Reynolds stress tensor can be simplified using the continuity equation $\nabla \cdot \vec{V}' = 0$:

$$\nabla \cdot (\vec{V}' \vec{V}') = \vec{V}' \cdot \nabla \vec{V}' + \vec{V}' \nabla \cdot \vec{V}' = \vec{V}' \cdot \nabla \vec{V}' \quad (5.45)$$

Using the following identities:

$$\vec{V}' \cdot (\vec{V} \cdot \nabla \vec{V}) = \vec{V} \cdot \nabla \left(\frac{\vec{V}' \vec{V}'}{2} \right) \quad (5.46)$$

$$\vec{V}' \cdot (\vec{V}' \cdot \nabla \vec{V}) = (\vec{V}' \vec{V}') : \nabla \vec{V} \quad (5.47)$$

The expression $\nabla \vec{V}$ is the sum of the deformation and rotation tensor and with turbulence kinetic energy defined as:

$$k = \frac{1}{2} \overline{\vec{V} \cdot \vec{V}} = \frac{1}{2} \overline{V^2} \quad (5.48)$$

Substituting into equation (5.44), gives:

$$\frac{\partial k}{\partial t} + \overline{\vec{V} \cdot \nabla k} + \overline{(\vec{V}' \vec{V}') : \nabla \vec{V}} + \overline{\vec{V}' \cdot (\vec{V}' \cdot \nabla \vec{V}')} = -\frac{1}{\rho} \overline{\vec{V}' \cdot \nabla p'} + \nu \overline{\vec{V}' \cdot \Delta \vec{V}'} \quad (5.49)$$

Simplifying and using Cartesian notation gives:

$$\frac{\partial k}{\partial t} + \overline{V_i} \frac{\partial k}{\partial x_i} = -\overline{V'_i V'_j} \frac{\partial \vec{V}}{\partial x_j} - \frac{\partial}{\partial x_i} \left[V'_i \left(k + \frac{p'}{\rho} \right) \right] + \nu \overline{V'_i} \frac{\partial^2 V'_i}{\partial x_j^2} \quad (5.50)$$

k-ε Models

The standard form of the ε transport equation is as used by Jones and Launder [88]:

$$\frac{\partial \varepsilon}{\partial t} + V_i \frac{\partial \varepsilon}{\partial x_i} = \frac{\partial}{\partial x_i} \left[\left(\nu + \frac{\nu_t}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_i} \right] + C_{\varepsilon 1} f_{\varepsilon 1} \frac{\varepsilon}{k} - C_{\varepsilon 2} f_{\varepsilon 2} \frac{\varepsilon^2}{k} \quad (5.51)$$

The production and dissipation terms of the dissipation equation are formed from the production and dissipation terms of the turbulent kinetic energy equation scaled by ε/k and multiplied by empirically determined constants and wall damping functions ($C_{\varepsilon 1} f_{\varepsilon 1}$ and $C_{\varepsilon 2} f_{\varepsilon 2}$). Moreover, an additional damping term is needed for the the eddy viscosity term in the $k - \varepsilon$ equations for near-wall cases so that k and ε will have proper behavior. The general form of the eddy viscosity relations for low Reynolds number models is:

$$\nu_t = C_\nu f_\nu \frac{k^2}{\varepsilon} \quad (5.52)$$

The wall boundary condition for turbulent kinetic energy is:

$$k_{wall} = 0 \quad (5.53)$$

As the turbulent kinetic energy is set to zero at the wall, the eddy viscosity is also set to zero at the wall. For dissipation the wall boundary condition maybe found from the turbulent kinetic energy equation:

$$\varepsilon_{wall} = \frac{\partial}{\partial x_i} \left(\nu \frac{\partial k}{\partial x_i} \right) \quad (5.54)$$

This boundary condition for ε can on occasion cause numerical stability problems, so Speziale suggests replacing it with:

$$\varepsilon_{wall} = 2\nu \left(\frac{\partial \sqrt{k}}{\partial x_i} \right)^2 \quad (5.55)$$

The turbulent kinetic energy is usually assigned to the free stream turbulence level at the free stream location if the value is known, if turbulence intensity or level is unknown, then the turbulent kinetic energy term is set to a small number at the free-stream boundary. The eddy viscosity is also set to a small number at the free-stream boundaries. The free-stream value of the turbulent dissipation may be calculated from the definition of the eddy viscosity. The coefficient values used in the standard k- ε model are listed in table 5.1

Table 5.1: Coefficients used in the standard k- ε model.

Coefficient	Value
$C_{\varepsilon 1}$	1.44
$C_{\varepsilon 2}$	1.92
σ_k	1.0
σ_ε	1.3
C_μ	0.09
κ	0.41

k- ω Model

The use of a ω transport equation was first introduced by Kolmogorov [85]. The model uses a transport equation that is associated with the smallest eddy and includes both ε and ω . Wilcox [89, 90] defined the ω term as:

$$\omega = \frac{\varepsilon}{C_\nu k} \quad (5.56)$$

The eddy viscosity definition for the Wilcox [89, 90] formulation is:

$$\nu_t = \frac{k}{\omega} \quad (5.57)$$

Using the Wilcox formulation , a transport model for ω can be written as:

$$\frac{\partial \omega}{\partial t} + V_i \frac{\partial \omega}{\partial x_i} = \alpha \frac{\omega}{k} \tau_{ij} \frac{\partial V_j}{\partial x_i} - \beta \omega^2 + \frac{\partial}{\partial x_i} \left(\left(\nu + \sigma \frac{k}{\omega} \right) \frac{\partial \omega}{\partial x_i} \right) \quad (5.58)$$

The term τ_{ij} is the specific Reynolds stress tensor. The coefficients for the ω transport equation are given in table 5.2. The wall boundary condition for k is similar to the $k - \varepsilon$

Table 5.2: Coefficients used in the k- ω turbulence model.

Coefficient	Value
α	$\frac{5}{9}$
β	$\frac{3}{40}$
σ	$\frac{1}{2}$

model with k set to zero at the boundary walls. In the case of ω , the definition of ω as stated in (5.56) suggests that at wall ω would tend to infinity as k is set to zero. Wilcox using asymptotic arguments, proposed that ω be set at the wall to :

$$\omega_{wall} = \frac{60\nu}{C_\mu (\Delta y^2)} \quad (5.59)$$

Here the term Δy is the normal distance to the first point from the wall. The recommended free-stream boundary conditions include, $\omega_\infty = 10 \frac{U_\infty}{L}$, $k_\infty = \nu_{t\infty} \omega_\infty$ and $\nu_{t\infty}/\nu_\infty = 0.001$. The selection of the free-stream ω is important as the model is highly sensitive to free-stream ω .

Shear Stress Transport (SST) Model

Generally the $k - \varepsilon$ models are more accurate in shear type flows and are well behaved in the far-field. However, $k - \omega$ models are more accurate and much more numerically stable in the near-wall region. Menter [60, 61] suggested a blended model that is a $k - \omega$ model near the wall and transitions to a k- ε model away from the wall. The model has the form:

$$\frac{\partial k}{\partial t} + V_i \frac{\partial k}{\partial x_i} = \frac{\partial}{\partial x_j} \left[(\nu + \sigma_k \nu_t) \frac{\partial k}{\partial x_i} \right] + P_k - \beta^* \omega k \quad (5.60)$$

$$\frac{\partial \omega}{\partial t} + V_i \frac{\partial \omega}{\partial x_i} = \frac{\partial}{\partial x_i} \left[(\nu + \sigma_\omega \nu_t) \frac{\partial \omega}{\partial x_j} \right] + \alpha D^2 - \beta \omega^2 + (1 - F_1) \frac{2\sigma_{\omega 2}}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i} \quad (5.61)$$

The blending function F_1 is defined as:

$$F_1 = \tanh \left\{ \min \left[\max \left(\frac{\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{y^2 \omega} \right), \frac{4\rho\sigma_{\omega 2} k}{CD_{k\omega} y^2} \right] \right\}^4 \quad (5.62)$$

The term y is the normal distance to the wall and $CD_{k\omega}$ is the positive portion of the cross diffusion term defined by:

$$CD_{k\omega} = \max \left(2\rho\sigma_{\omega 2} \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i}, 10^{-20} \right) \quad (5.63)$$

The production term P_k is defined as:

$$P_k = \min \left(\tau_{ij} \frac{\partial V_i}{\partial x_j}, 10\beta^* k\omega \right) \quad (5.64)$$

The coefficients σ_k , σ_ω , γ , and β are computed using the general form:

$$\phi = F_1 \phi_1 + (1 - F_1) \phi_2 \quad (5.65)$$

In this instance ϕ_1 corresponds to the coefficients from the $k-\omega$ model and ϕ_2 corresponds to coefficients of the $k-\varepsilon$ model. For example the α closure coefficient is blended form of α_1 and α_2 , similarly, β is a blended form of β_1 and β_2 . The values of the closure coefficients are listed in table 5.3. The eddy viscosity is calculated from:

$$\nu_t = \frac{a_1 k}{\max(a_1 \omega, \omega_v F_2)} \quad (5.66)$$

The term Ω_v in this case is the vorticity obtained from $\nabla \times \vec{V}$. F_2 is defined using:

$$F_2 = \tanh \left\{ \max \left(\frac{2\sqrt{k}}{\beta^* \omega y}, \frac{500\nu}{y^2 \omega} \right) \right\}^2 \quad (5.67)$$

The first term in the denominator of (5.66) comes from the traditional definition of eddy viscosity for a $k-\omega$ model, and the second term in the denominator represents an attempt

Table 5.3: Coefficients used in the SST turbulence model.

Coefficient	Value
α_1	$\frac{5}{9}$
α_2	0.44
β_1	$\frac{3}{40}$
β_2	0.0828
β^*	$\frac{9}{100}$
σ_{k1}	0.85
σ_{k2}	1
$\sigma_{\omega 1}$	0.5
$\sigma_{\omega 2}$	0.856

to improve the performance of the model for adverse pressure gradients and is based on a relationship for shear stress in a boundary layer. The second term in the denominator of (5.66) reduces the eddy viscosity in the logarithmic and wake region of the boundary layer when adverse pressure gradients are present. The coefficients used in the SST model are listed in table 5.3. The boundary conditions used for the SST model are similar to the boundary conditions used by the $k\text{-}\omega$ model.

6. EXPERIMENTAL FACILITY

In this chapter brief overview of the experimental test bed used at the TPFL. More details can be found in Schobeiri et al. [91, 92, 93]. The high pressure model turbine facility at the TPFL is shown in figure 6.1. It consists of nine main components, a 300hp electric drive for the compressor unit, a compressor unit directly coupled to the electric motor drive, a venturi mass flow meter, a honeycomb diffuser unit, the three stage research turbine, inlet air heater unit, torque meter and dynamometer unit, and a probe traversing unit.

6.1 Compressor and Electric Drive Unit

A 300 hp electric motor is used to drive a three stage centrifugal compressor supplying air with a maximum pressure difference of 8 psi and a maximum volumetric flow rate of $18,800 \text{ ft}^3/\text{min}$. The compressor is set to operate in suction mode and its pressure and volume flow can be varied by the frequency controller from 0 to 62 Hz . The compressor unit is located in a housing outside the test cell and is connected via piping to the venturi mass flow meter. The suction operation mode means the absolute pressure throughout the test facility is below atmospheric pressure. The inlet to the turbine is at room temperature, due to the expansion process the temperature at the compressor inlet can become low enough to create condensation which therefore mandated the need for the inlet heater to the unit to bring up the inlet temperature to about 50°C , giving a compressor inlet temperature of 24°C . At the exit of the three stage turbine the flow exhibits a high swirl characteristic and therefore a straightening mechanism is required to allow for more accuracy in the mass flow rate readings obtained from the venturi mass flow meter. Therefore, a honeycomb and flow straightener are placed at the exit of the turbine and inside the pipe contraction as shown in figure 6.1.

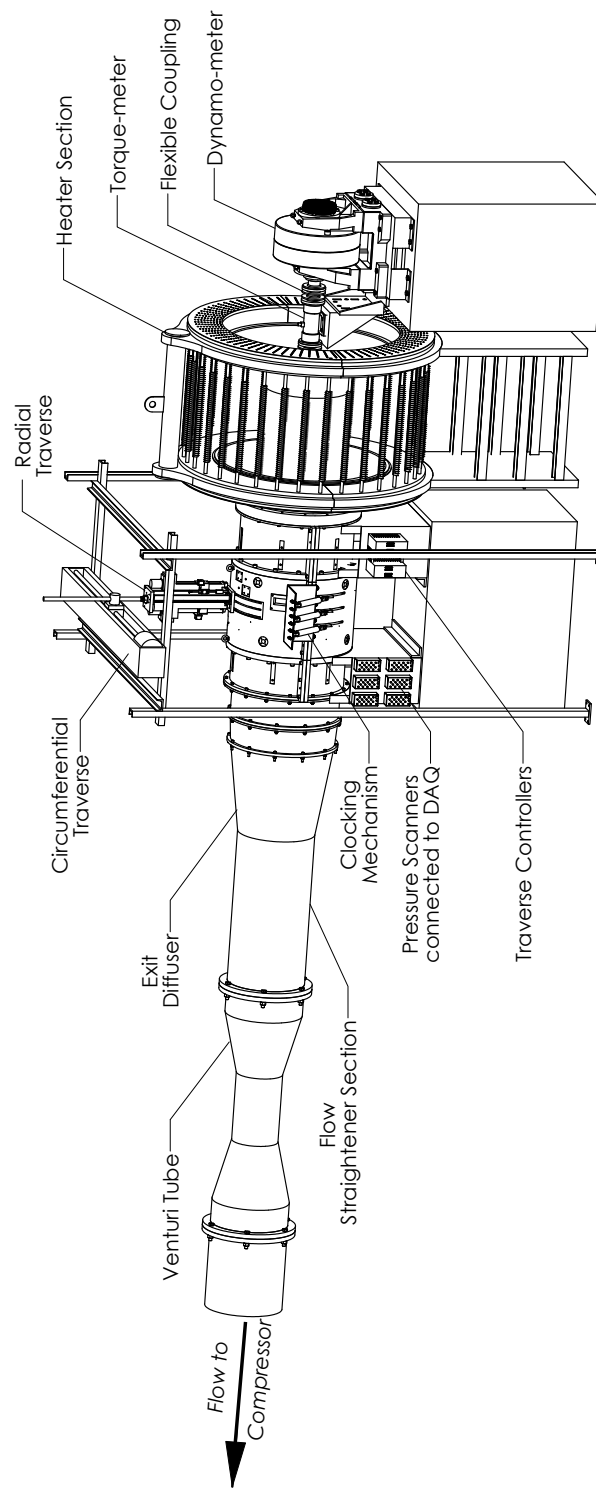


Figure 6.1: Multistage model turbine at Texas A&M University's TPFL as shown in Schoeiri et al. [2].

6.2 Multi-Stage Turbine Engine

The engine is a multi stage air turbine with the dimensions and specifications shown in table 6.1. The engine was built in 1999 for the specific use of blade performance assessment. The turbine was employed to measure flow angles and velocities, pressures, and efficiency at multiple positions. The design is highly versatile and employs the use of casings with stator rings, three T-rings for sealing the 90-degree circumferential traversing slots. The initial engine setup incorporated shrouded rotor blades which were attached to the rotor cylinder, which is connected to the shaft through two taper locking mechanisms.

As explained in Schobeiri et al. [2, 3, 91, 92, 93] the turbine has undergone many different design modifications including adapting film cooling as well as platform cooling locations at the first rotor row, changing the turbine from a three stage to two stage configuration as well as adding a plexiglass view port that allows for pressure sensitive paint (PSP) and temperature sensitive paint (TSP).

Table 6.1: Turbine research facility specifications.

Item	Specification
Stage number	$N = 2-3$
Hub radius	$r_{\text{hub}} = 280.0 \text{ mm}$
Tip radius	$r_{\text{tip}} = 343.5 \text{ mm}$
Blade height	$H = 63.5 \text{ mm}$
Power	$P = 26 - 96 \text{ kW}$
Mass flow	$\dot{m} = 2.56 - 3.84 \text{ kg/sec}$
Speed range	$n = 1750 - 3000 \text{ rpm}$
Pressure ratio	$PR = 1.16 - 1.45$

6.3 Inlet Air Heater

As stated earlier an annulus type heater is used to bring the temperature of the inlet airflow to about 50°C in order to avoid condensation at turbine exit. The air inlet temperature is measured using a K-type thermocouple and this reading is fed into a controller

which keeps the heater at the required inlet temperature at all times.

6.4 Torque Meter and Dynamometer

The turbine is not directly coupled to the compressor and in order to continuously apply load to the unit and prevent shaft overspeed, a Froude Consine dynamometer was coupled to the shaft via a Rex Nord flexible coupling. A Himmelstein dynamometer is also coupled to the shaft and is used for the accurate measurement of the torque applied on the rotor shaft. The torque meter is a C-enhanced meter with a maximum torque rating of 6000 lb-in and allows for $\approx 0.02\%$ of full scale load accuracy. The dynamometer is an eddy current type unit that is run in a constant speed mode and can be remotely controlled using a Texcel V4-EC controller. The dynamometer has a maximum speed rating of 8000 rpm and can apply a maximum torque of 500 $N.m$, it is used to measure the shaft speed at an accuracy of 1 rpm and can additionally measure the shaft torque with an accuracy of 1.25 $N.m$.

6.5 TPFL Instrumentation

6.5.1 Probe Traversing Unit

The traversing unit is used for circumferential and radial traverse within the engine, and allows for a wide data acquisition window and high data resolution. It is designed to traverse at 3 stations, stations 3, 4 and 5 as shown in figure 6.1, this is possible through the use of sealing T-rings. The unit employs the use of 2 Whedco controllers each able to control 2 axes, 3 axes are radial movement at stations 3, 4 and 5 and the 4th axis is used to control the circumferential movement of the whole traverse. Figure 6.1 shows the mechanism used to create the circumferential movement of the traverse. The traverse in the circumferential direction can allow for 4 blade spacings as a maximum, and radially moves the 3 probes from hub to tip with a 1 mm step displacement as a maximum spatial resolution. Interfacing with the movement controllers is via a LABview program which is also used to acquire the pressure and temperature readings.

6.5.2 Engine Monitoring Instrumentation

The TPFL model turbine unit utilizes several monitoring equipment to ensure the unit is run within allowable safety limits. The equipment include vibration velocity meters, and temperature monitoring thermocouples. The vibration monitoring equipment are velocity probes of the type Bentley-Nevada PN 330500 (100 mV/in/s) and are mounted on the front and back shaft bearings at 6 and 12'o clock positions. A fifth probe is placed at the base of the torquemeter support plate. The output from the five probes is collected using two Bently-Nevada 3300/55 dual velocity monitoring system which give a minimum observable value of 0.05 in/s . Temperature monitoring thermocouples are placed to monitor critical components within the engine. The thermocouples are J-type thermocouples with 50 ft of extension grade thermocouple wire. The temperature is monitored at the shaft bearings, venturi inlet and at the turbine inlet. The bearings have two thermocouples placed at 6 and 12 o'clock positions, and are used to make sure the shaft bearing temperatures do not exceed 70°C . For the bearings the use of temperature and vibration level readings act as a simple means to monitor bearing health and determine the need to change bearings. At the venturi inlet, due to the concern for condensation a thermocouple is placed to monitor the temperature of airflow exiting the turbine. The temperature at this point is monitored to make sure it does not go below 21°C as this would signify condensation. Turbine inlet temperature is monitored and controlled using a feedback loop on the heater controller. The inlet temperature to the unit is an important parameter for each experimental run and therefore active control of this parameter is necessary. Additionally, the temperature is monitored to make sure it does not exceed safety limits. The output from thermocouples is fed into a six channel temperature monitoring system.

6.5.3 Turbine Performance Instrumentation

The pressure and temperature measuring equipment used to obtain engine performance curves include wall static pressure taps, and exit and inlet rakes. Four combined total pressure and total temperature rakes were placed upstream of the first stator row. They were

located radially at 2, 4, 8 and 10 o'clock positions. Each rake consists of 4 total pressure and 3 total temperature probes. The total pressure probes are simply Pitot tubes, and the total temperature probes are calibrated J-type thermocouples, with their calibration implemented into the performance reduction program. To reduce the wake thickness originating from the trailing edge of the inlet rakes, the rakes were shaped aerodynamically with a round leading edge and a sharp trailing edge. Four combined total pressure and total temperature rakes were placed downstream of the last rotor row. Wall static pressure taps were placed circumferentially at stations 3 and 4 on the sealing T-rings. Each ring had two taps at 11 and 1 o'clock positions. The total number of stationary instrumentation requires 78 measurement channels.

Three L-shaped five hole probes were used in the traverse unit to resolve pressure contours, velocity components, flow angles and spanwise total pressure loss coefficients. Each probe was calibrated with a non-nulling technique in a pitch-yaw axis calibration facility at velocities similar to what is exhibited within the engine stations. The non-nulling calibration used to calibrate the five hole probes required independently placing each probe onto an angular indexing mechanism which automatically pitched the probe through the range of angles (-25° to 25° in pitch and yaw), with 1 degree increments. The probe indexing mechanism utilizes two stepper motors which are computer controlled to produce fine angle resolution.

6.6 Data Acquisition

Calibration procedures were required for the torquemeter, dynamometer, and the pressure transducers. Frequent calibration of the above equipment ensures that the best possible data can be obtained. The torquemeter is calibrated electronically by matching specified internal coefficients in the controller unit. The torque reading from the dynamometer was calibrated using weights applied on the unit's casing. A process of loading and unloading 100 kg in 10 kg increments was used and an accurate calibration slope is obtained and stored internally in the dynamometer controller unit. For data acquisition purposes the torquemeter outputs a 0-10 V signal that is proportional to the measured torque, this

signal requires a second calibration in order to accurately correlate the measured output voltage with the measured torque. This second calibration was also required of the output voltage obtained from the dynamometer which was proportional to the shaft speed. The seven pressure scanners have internal computer controlled temperature compensation and are designed to maintain an accurate slope of the calibration over time. The sensors are subject to a small zero-drift, which requires a zeroing procedure prior to any data acquisition or engine startup. The absolute pressure transducer utilizes a very accurate pressure diaphragm in addition to internal temperature compensation. Data acquisition hardware consists of a National instruments (NI) AT-MIO-16E01 12 bit analogue to digital (A/D) converter with a sampling rate of up to 1.25 MS/s. The pressure scanners were seven differential pressure scanners and one absolute pressure transducer of PSI Model 9032. The scanners have a long time accuracy of 0.02%, and have a measurement resolution of $\pm 0.003\%$ of full scale for sensor range of less than 2.5 psi, And a resolution of $\pm 0.05\%$ of full scale for sensor range of greater than 2.5 psi. The differential pressure scanners and absolute pressure transducers were all referenced to a single open system in order to dampen out any pressure spikes that may occur. For temperature measurement, two Fluke 2640A NetDAQ networked data acquisition systems are used for recording the temperatures within the engine. Each unit is capable of recording up to 20 channels at a maximum rate of 143 samples per second. The accuracy of J-type thermocouples measurements are $0.80^{\circ}C$ with a resolution of $0.03^{\circ}C$. The thermocouples are connected using extension grade thermocouple wire which has a limit of error of $\pm 2.2^{\circ}C$. The engine component signals such as pressure, temperature, vibration, torque and rotational speed are transferred to a data acquisition computer via the aforementioned instruments. A continuous data acquisition during the engine operation is done through a LABview interface which writes the data to the desired output file. The data is then processed in the main data reduction and analysis computer using a Fortran code.

7. TWO-STAGE TURBINE STUDY*

In this chapter* a two-stage axial turbine blade design is analyzed numerically at steady and unsteady conditions, compared to experimental data in order to evaluate the design at both design and off-design operating conditions. A new 2-stage rotor was designed, manufactured and installed into the turbine facility at the Turbine Performance and Flow Research Laboratory's (TPFL), having been converted from an earlier 3-stage configuration. The specifications for the TPFL are shown in table 6.1 and figure 6.1, the 2-stage configuration's specifications are listed again in table 7.1, and include equal blade counts for first and second stage stators at 66 blades, and equal blade counts for both first and second stage rotor blades at 63. Additionally, with a lighter rotor the turbine can now be

Table 7.1: Two-stage turbine research facility specifications.

Item	Specification
Stage number	$N = 2$
Hub radius	$r_{\text{hub}} = 280.0 \text{ mm}$
Tip radius	$r_{\text{tip}} = 343.5 \text{ mm}$
Blade height	$H = 63.5 \text{ mm}$
Blade number	Stator 1, 2 = 66
Blade number	Rotor 1, 2 = 63
Power	$P = 26 - 96 \text{ kW}$
Mass flow	$\dot{m} = 2.56 - 3.84 \text{ kg/sec}$
Speed range	$n = 1750 - 3000 \text{ rpm}$
Pressure ratio	$PR = 1.16 - 1.45$

operated at a rotational speed of up to 3000 rev/min versus the earlier three stage configurations that were limited to a maximum of 2800 rev/min. A detailed view of the 2-stage

*Part of the data reported in this chapter is reprinted with permission from "Investigating the Cause of Computational Fluid Dynamics Deficiencies in Accurately Predicting the Efficiency and Performance of High Pressure Turbines: A Combined Experimental and Numerical Study" by M.T. Schobeiri, S.A. Abdelfattah and H.C. Chibli, 2012. ASME Journal of Fluids Engineering, 134(10), 101104-101104-12, Copyright 2012 by American Society of Mechanical Engineers.

geometry is shown in figure 7.1 showing both a blade-to-blade view and a meridional view of the geometry. The design adds the use of knife edge seals at the rotor shroud, as well as seals at the stator blade hub. Detail drawing sof the stator and rotor design are shown figures 7.2 and 7.3 respectively, both figures are obtained from [10].

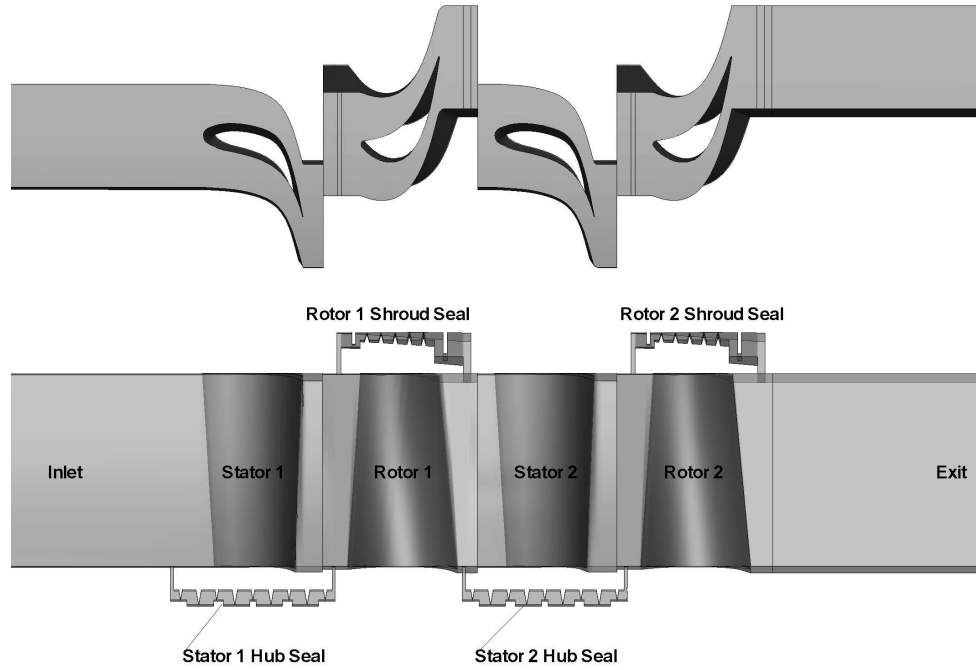


Figure 7.1: Two-stage turbine geometry as shown in [2].

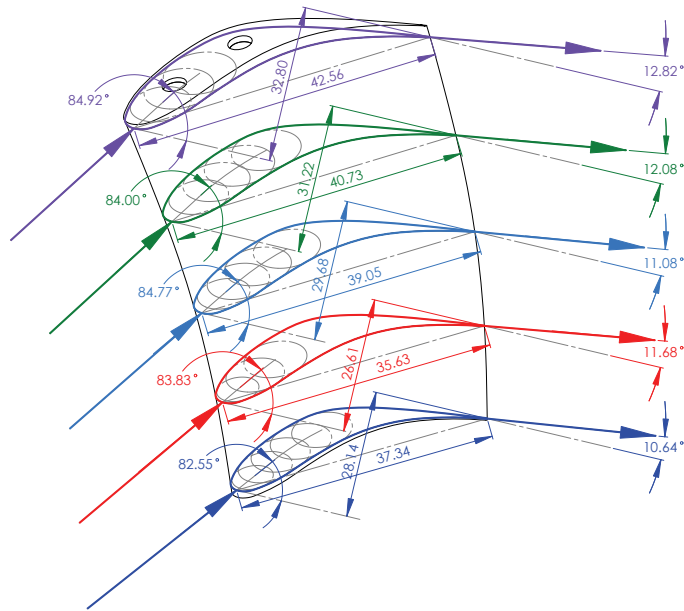


Figure 7.2: Two-stage turbine stator blade geometry details.

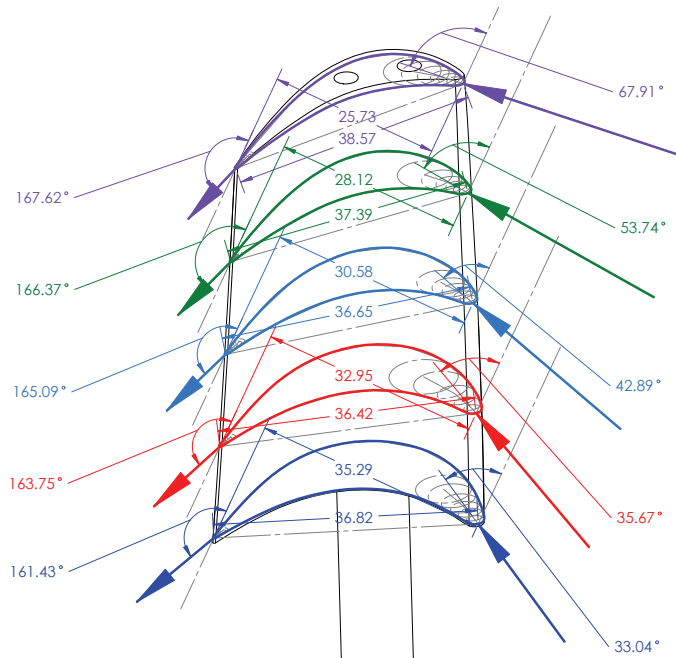


Figure 7.3: Two-stage turbine rotor blade geometry details.

7.1 Numerical Simulation

The first step required for numerical analysis is generating a adequate numerical discretization or mesh of the geometry. The mesh needed should satisfy:

- Solver minimum acceptable skew values.
- Solver minimum acceptable aspect ratio.
- Solver acceptable mesh quality usually observed via 3x3x3 determinant or minimum angles.
- Based on turbulence model used wall Y^+ values should be reasonable (low-Re models Y^+ of about 1 are needed).

Moreover, the generated mesh can at times be limited by the computational resources available for analyses, so an important balance between computational resources and required accuracy must be evaluated by the users. Lastly and most importantly, the solution should show mesh independence, and this is usually achieved by gradually refining the mesh until the observed solution shows no change in specific results such as turbine predicted efficiency, flowrate, and blade exit velocity/pressure profiles.

7.1.1 Mesh Generation

Mesh generation was performed using the ICEM-CFD grid generator. An annular representation of the geometry was imported into the ICEM-CFD and a grid was generated using the multi-block meshing technique, which generates a mapped structural hexahedral mesh that is based on the specified blocking topology. An O-grid topology was used in regions adjacent to walls such as the blade pressure and suction surfaces in order to create a boundary layer refinement next to the walls. In each O-grid a minimum of 16 node points were used to represent the boundary layer, and a first grid point was estimated (and later shown) to provide Y^+ values of 1 or less. Five grids were generated through a coarsening process of the fine mesh to show mesh independence, to maintain reasonable aspect ratios

Y^+ values of 1 were not possible for the much coarser grids. Figure 7.4 shows one of the numerical grids that were used for the analyses and as part of the mesh sensitivity study. A summary of the 5 mesh densities generated are summarized in table 7.2.

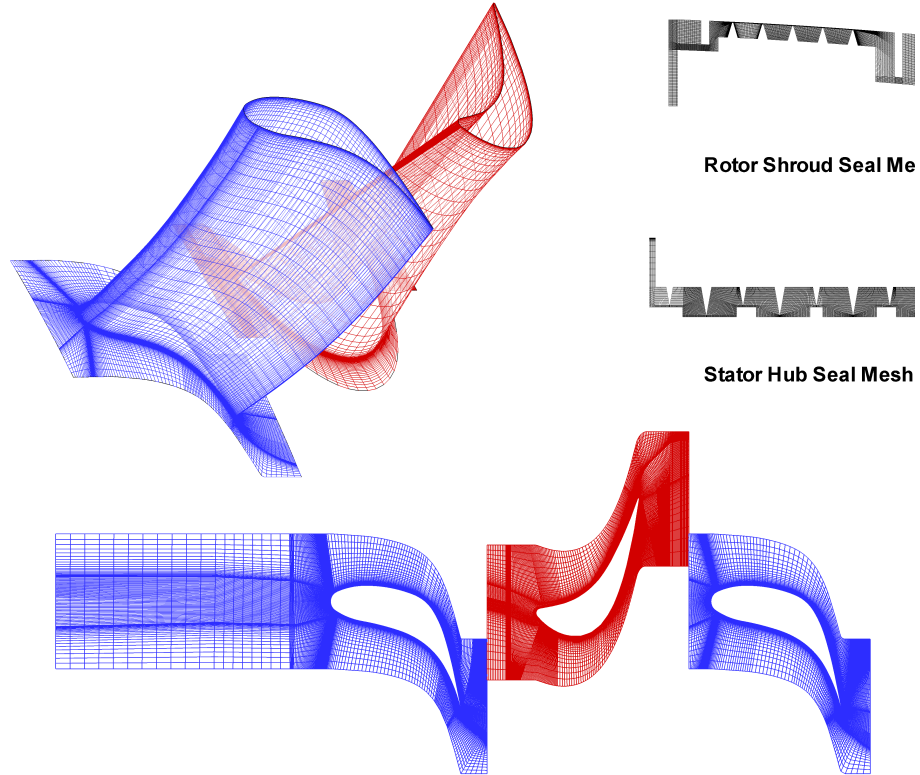


Figure 7.4: Interstage analysis numerical mesh (mesh4).

7.1.2 Boundary Conditions

The boundary conditions used in both steady numerical simulations as well as unsteady numerical simulations are based on experimental data, and some of the boundary conditions used are listed in table 7.3. The inlet location's data is based on various inlet rakes placed

after the heater that measure both pressure and temperature. The domain exit conditions are based on exit rakes that provide both pressure and temperature conditions at the turbine exit. Between rotating and non-rotating domains a interface was used which, for unsteady analyses was a sliding mesh interface, whereas in steady analyses a mixing plane model as used by [30] was employed. The mixing plane method allows the user to have a mismatch in patch area between the upstream blade row exit patch and the downstream inlet patch, however in the case of the sliding mesh this is not possible, and as discussed in the URANS results the blade numbering was slightly modified in order to satisfy an equal patch area across the sliding mesh interface. The inlet turbulence intensity was based on experimental testing conducted in the previous tests at the TPFL and was approximated to 2%.

7.1.3 Turbulence Model

The turbulence model used in this study was the Shear Stress Transport $k - \omega$ model (SST) which uses a blending function at the wall location. Therefore, great care was taken to create a highly refined discretization at the wall boundaries. Introduced by Menter [60] and [61], the SST model has proven itself as a popular turbulence model for turbomachinery applications, with its ability to combine both qualities of the $k - \varepsilon$ and $k - \omega$ models. As per Menter [61] the transformation of the original Wilcox model is done by using a blending function F_1 and applying a $1 - F_1$ term to the $k - \varepsilon$ model, The factor F_1 allows for the

Table 7.2: Mesh densities used in the two-stage turbine numerical analyses.

Mesh Designation	Total Number of Hexahedral Elements
1	575763
2	1240949
3	2113746
4	4991874
5	8566769

Table 7.3: Two-stage turbine boundary conditions for 1800 rpm and 2600 rpm at a pressure ratio of 1.38.

Boundary Condition	1800 rpm	2600 rpm
Reference Pressure [kPa]	101.261	100.975
Total Pressure Inlet [kPa]	100.496	100.271
Static Pressure Outlet [kPa]	71.654	72.021
Total Temperature Inlet [K]	323.48	324.0
Total Temperature Outlet [K]	298.3	297.05
Inlet turbulence Intensity[%]	2	2

smooth transition between the modified $k - \varepsilon$ and the $k - \omega$ models depending on whether computation location is inside or outside the boundary layer. The basic equations for the SST turbulence model are listed in 5.2.3 as per [60] and [61]

7.1.4 Grid Independence

The design case was run using the various grid discretizations that are listed in table 7.2. All cases were converged to satisfactory convergence criteria and a comparison was made of various performance parameters as well as flow parameters as a function of mesh size. Moreover comparing various interstage circumferentially averaged flow parameters illustrates the difference between the coarser mesh1 and mesh2, versus the finer grids that were used. As shown in figure 7.5, the coarser mesh1 and mesh 2 exhibit a significant difference in efficiency prediction than the finer discretizations of mesh 3 to 5. The total-to-static efficiency is defined as (where state 1 and 2 represent the inlet and exit conditions respectively, and 2s represents the exit isentropic condition):

$$\eta_{ts} = \frac{H_1 - H_2}{H_1 - h_{2s}} \quad (7.1)$$

For an isentropic expansion this can be calculated using:

$$\eta_{ts} = \frac{1 - \frac{T_{t1}}{T_{t2}}}{1 - \left[\frac{P_{s2}}{P_{t1}} \right]^{\frac{k-1}{k}}} \quad (7.2)$$

In the case of total-to-total efficiency the expression becomes:

$$\eta_{tt} = \frac{H_1 - H_2}{H_1 - H_{2s}} \quad (7.3)$$

For an isentropic expansion this can be calculated using:

$$\eta_{tt} = \frac{1 - \frac{T_{t1}}{T_{t2}}}{1 - \left[\frac{P_{t2}}{P_{t1}} \right]^{\frac{k-1}{k}}} \quad (7.4)$$

Figures 7.6 to 7.10 shows various circumferentially averaged parameters versus normalized at stations 3 and 4 with the 5 different mesh discretizations used. Parameters compared include total pressure, Mach numbers, and turbulence kinetic energy. Although from the prior performance based figures mesh 3 shows a good compromise and the begin of mesh independence, mesh 3 does not capture the large gradients in Mach number at hub and tip locations, due to secondary flows as well as mesh 4 and mesh 5 do. However this seems to minimally effect the overall efficiency as shown in figure 7.5 where mesh 3 is the real start of grid independence. As a result going forward mesh 4 was used for the RANS and URANS based analysis for this blade design as well as other blade designs discussed in this document.

7.1.5 Steady (RANS) and Unsteady (URANS) Analyses

Using the generated grids, both a steady or RANS based analysis as well as a unsteady or URANS based analysis were run using the solver ANSYS CFX. The RANS cases were run under varying RPM and massflow rate (controlled by varying the inlet and exit pressure imposed). Both cases employed the Shear Stress transport model as briefly discussed in section 5.2.3. A major difference between steady and unsteady cases was the use of interface

models between each blade row's exit patch and the subsequent downstream inlet patch. In both analyses a high resolution upwind scheme was used in space, for the unsteady analysis a backward Euler second order time scheme was used for integration of the URANS equations.

Steady Analysis

In the steady model, the mixing plane mode, as explained by [30] was employed. Mixing plane interfaces are used to allow for steady calculations of flow through multistage blade rows that are rotating relative to each other. At a fixed axial location for each spanwise point, flow upstream of the mixing plane is assumed to mix out to a flow that exhibits pitchwise uniform enthalpy and entropy. This pitchwise uniform flow is then used as a inlet condition to the next downstream blade row. With any mixing process the resultant entropy is increased, where ideally this mixing should conserve mass, energy and momentum. The assumption then continues that this added mixing loss at the interface is similar to the

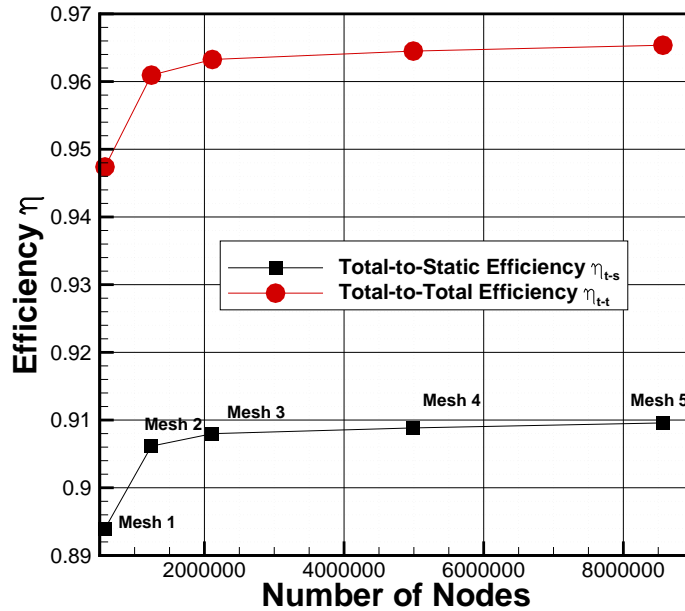


Figure 7.5: Total-to-static and total-to-total efficiencies as a function of node number.

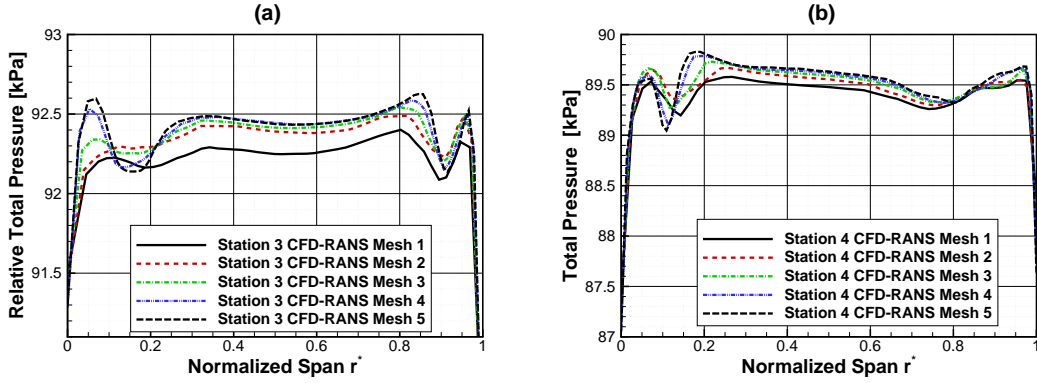


Figure 7.6: Comparison of circumferentially averaged relative total pressure and total pressure versus normalized span using various mesh densities at (a) station 3, and (b) station 4.

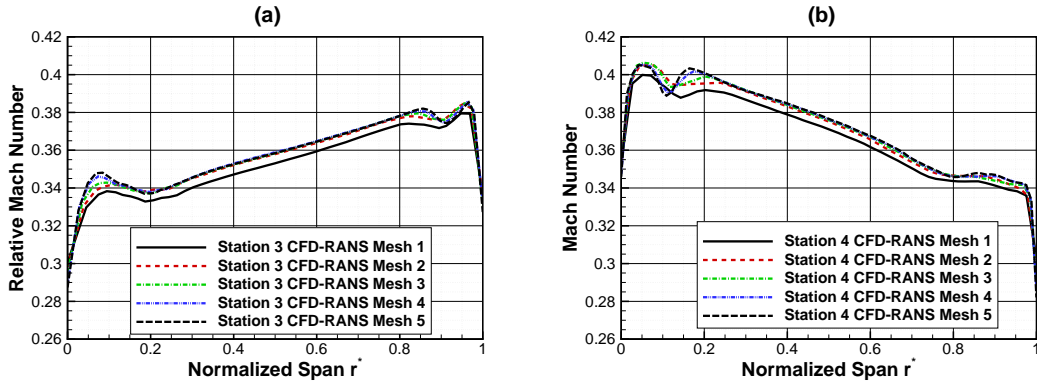


Figure 7.7: Comparison of circumferentially averaged relative Mach number and absolute Mach number versus normalized span using various mesh densities at (a) station 3, and (b) station 4.

loss generated when the flow mixes out in a realistic unsteady environment as explained by Denton in [4]. The mixing plane method allows for non-matching patch areas across the interface, and therefore the blade numbering that is physically used was maintained whilst using one blade pitch for each rotor and stator rows to model the turbine. In the highly off-design case where the machine pressure ratio (PR) was about 1.15 and the shaft speeds were as low as 1800rpm, a reduction in the 'false time' step used by the RANS solver was needed.

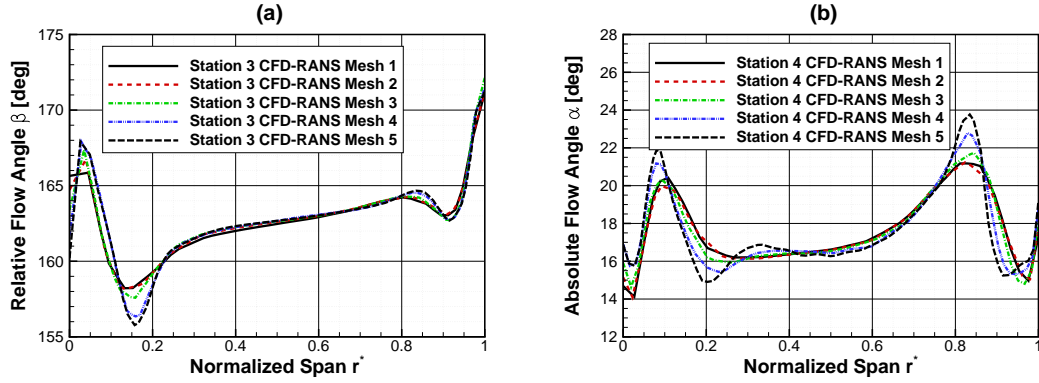


Figure 7.8: Comparison of circumferentially averaged relative flow angle β and absolute flow angle α versus normalized span using various mesh densities at (a) station 3, and (b) station 4.

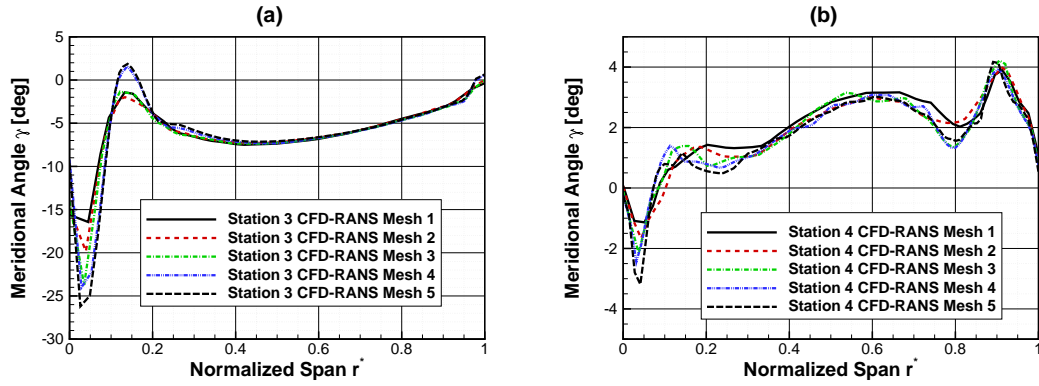


Figure 7.9: Comparison of circumferentially averaged meridional flow angle γ versus normalized span using various mesh densities at (a) station 3, and (b) station 4.

Unsteady Analysis

One way to avoid the use of mixing plane is to run an unsteady or a URANS analysis that employs a sliding mesh interface between the blade rows. The sliding mesh interface models the relative rotation between blade rows, and maps at each time instant the mass, energy and momentum profiles to the downstream blade row whilst accounting for relative motion between them. Unsteady calculations with sliding mesh are extremely time and hardware consuming, and a one to one interface is needed between blade rows. Meaning,

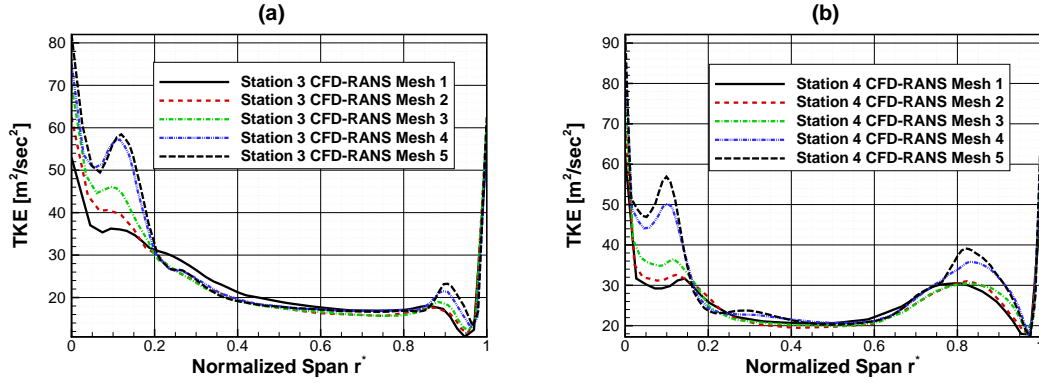


Figure 7.10: Comparison of circumferentially averaged turbulence kinetic energy k versus normalized span using various mesh densities at (a) station 3, and (b) station 4.

that the modeled sector between blade rows has to have a one to one area match. In the case of the two-stage configuration, this was difficult to achieve and so the stator blade numbers were reduced to 63 from the actual 66 in order to achieve a one-to-one area match with the rotor blades whilst using one blade pitch for the model. Due to the high computational cost of URANS analyses, only two cases were run which include a design case with 3000 RPM and a high pressure ratio as well as an off-design case that uses a low 1800 rpm shaft speed. For each case, the time step was selected to allow for a normalized pitch motion of approximately 0.025 for each time steps, or roughly 40 time steps over one blade passage period. This translates to a time step of 1.25×10^{-6} for the 3000 RPM case. The cases were run without storage of any flow data for about 10 blade passage periods after which the flow data was stored for each time step. Ten inner iterations were used by the CFX solver at each time step with a convergence residual error target of 1×10^{-6} .

7.2 Analysis of Numerical Results

In this section a comparison of the results obtained from the experimental versus both RANS and URANS based analysis using the SST model are presented. The main points of this section include the differences observed between design condition models as well as off-design (low flowrate) conditions, comparison of the numerical data with the experimentally

measured data, seal leakage flow effects. Additionally, a quick look at tip flows in the case of a design using various blade clearances is presented.

7.2.1 Seal Leakage

The seal domains present an interesting challenge to the gridding process. Mainly since some of the seals are considered rotating domains (rotor hub seals) and some are non-rotating (stator shroud). The interfacing between both the main blade domains and the seals would therefore require the use of mixing planes when there is a relative motion between them. This creates another approximation or error source to the model, the patches between the interface use a grid-to-grid interface interpolation if the mesh is non-matching. The massflow in the seals was found to be between 0.2% to 0.3% of the machine massflow. Comparing between a model using the seals domains and a model not accounting for the seals flows gives a difference of about one efficiency points at high shaft speeds and about 2 points at low shaft speeds. This is a substantial difference and as such the seals were used in all the RANS cases. In the URANS models, due to computational resource restrictions the seals were not included in the simulation. Figures 7.11 to 7.13 both show meridional absolute total pressure as well as the mixing of seal flow with the main passage flows which generates an extra source of loss.

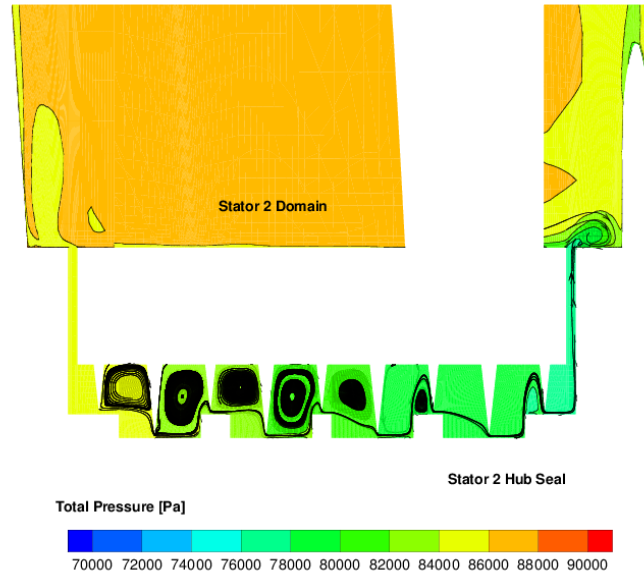


Figure 7.11: Meridional contour of total pressure in the stator 2 hub seal.

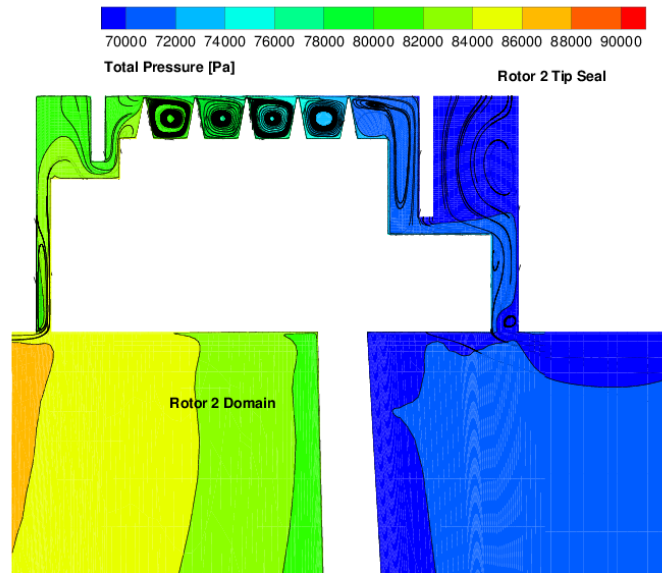


Figure 7.12: Meridional contour of total pressure in the rotor 2 tip seal.

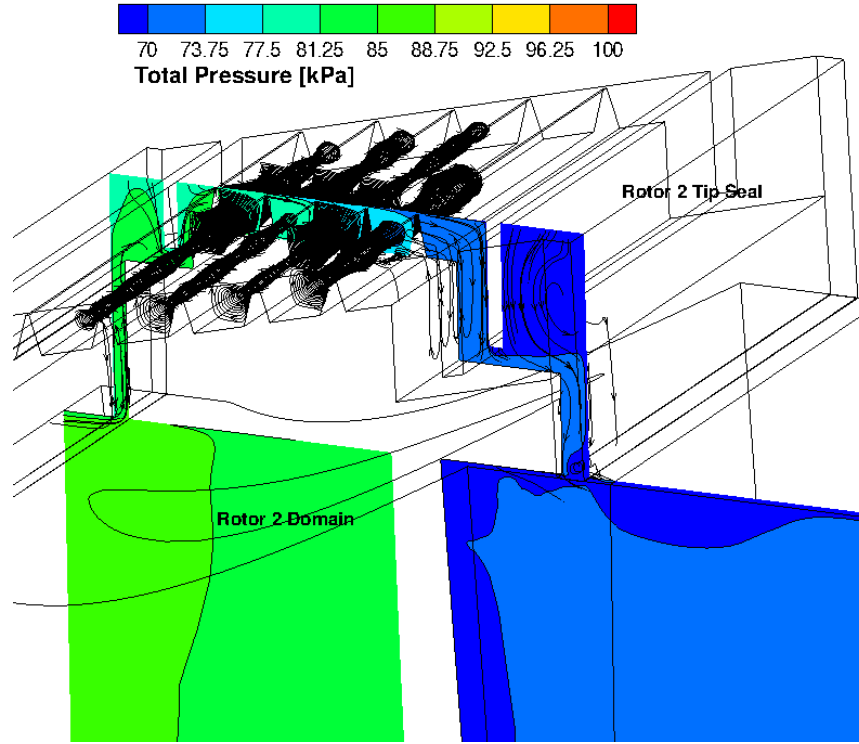


Figure 7.13: 3D visualization of leakage flow in rotor 2 tip seal.

7.2.2 Flow Visualization: Design and Off-design Conditions

An important point to consider is how changing the operating point via massflow changes or shaft speed changes, impact both the flow inside the model and the resulting changes in performance as a result of the operating point change. Although, most power generation turbomachinery are designed to operate at their design points, load changes and or start up shut down conditions present operating points that are not at an optimum for the specific design.

Figures 7.14 and 7.17 show flow visualization results for the second stage rotor at the high shaft speed/high massflow condition.

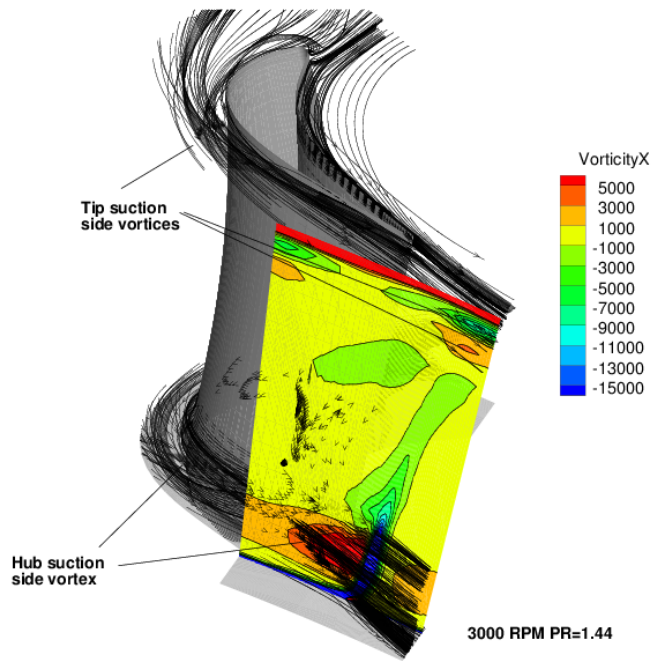


Figure 7.14: Two-stage turbine, stage 2 rotor hub and tip vortices for 3000 RPM PR=1.44 case.

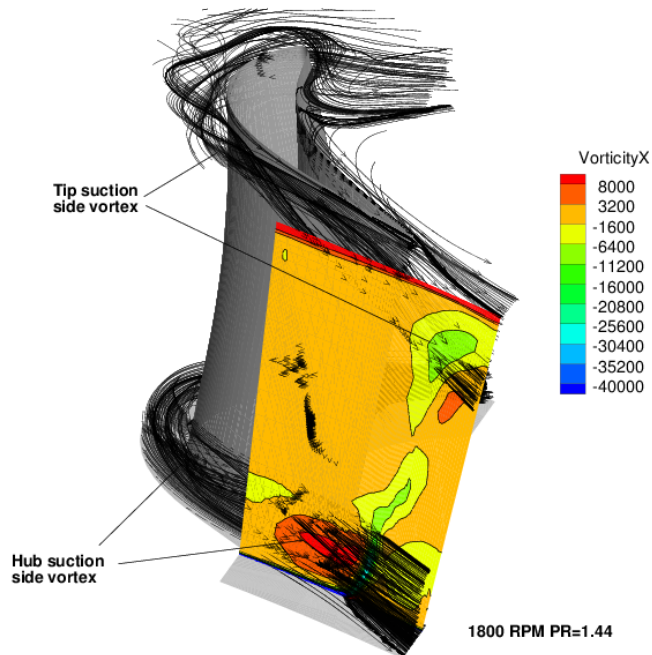


Figure 7.15: Two-stage turbine stage 2 rotor hub and tip vortices for 1800 RPM PR=1.44 case.

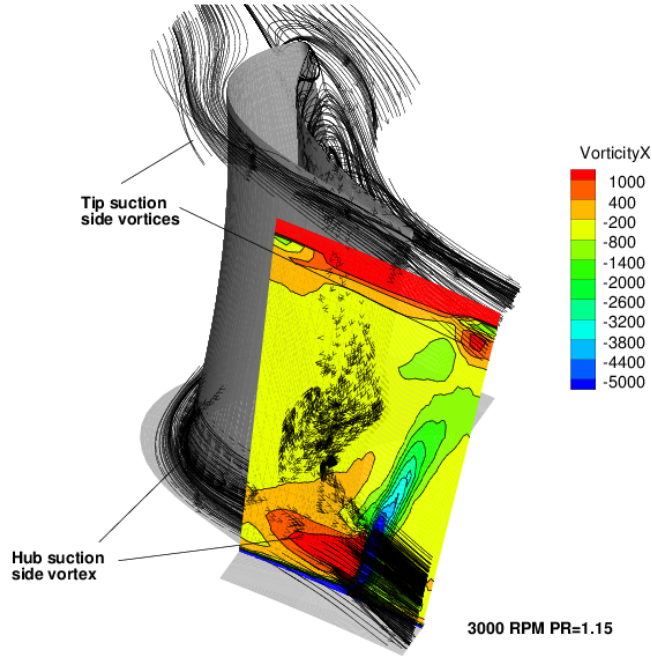


Figure 7.16: Two-stage turbine stage 2 rotor hub and tip vortices for 3000 RPM PR=1.15 case.

At the tip region for this operating point it can be seen that the flow incidence is nearly zero, but pronounced suction side vortices are formed at the hub and tip near wall locations. The hub vortex rotating in a counter-clockwise direction, whereas the tip vortex smaller in magnitude and rotating in a clockwise direction. At lower shaft speeds, a pronounced negative incidence is observed at the tip, and as expected the suction side tip and hub vortex show much large magnitudes. Additionally as shown in figures 7.15 and 7.18, a pressure side tip vortex is observed that mixes with the suction side vortex as well as the trailing edge wake is observed. On the other hand maintaining a high shaft speed at with low massflow or pressure ratio shows a high positive incidence on the stage 2 rotor blade, as observed in figure 7.16 with lower vorticity values when compared to the 1800 RPM case.

The high positive incidence generates a large separation zone at the pressure side, particularly near the blade meanline region, this is clearly shown in figure 7.19 where a large

separation region appears at approximately mid chord and extends from the near tip to hub regions. The separation at the off-design case can be further illustrated using a blade loading chart at 50% and 90% immersion for both the design and off-design conditions, which are shown in figures 7.20 and 7.21, moreover, figures 7.22 and 7.23 show vector plots and Mach number contours for the design and off design conditions where a large positive incidence creates a separation region on the pressure side surface. This is also shown in the blade loading chart as a constant C_p value, since when separation occurs there is no work being done by the blade.

In summary changing the operating point via reducing shaft speed or massflow generates detrimental flow conditions such as high positive incidence, which in turn creates large separation zones, and a high negative incidence which increases the intensity of suction side hub and tip vortices and causing high secondary flow losses. The 3000 RPM/1.44 PR case shows to be a near-optimum operating point as shown in figure 7.17, where there is little incidence at hub and tip locations.

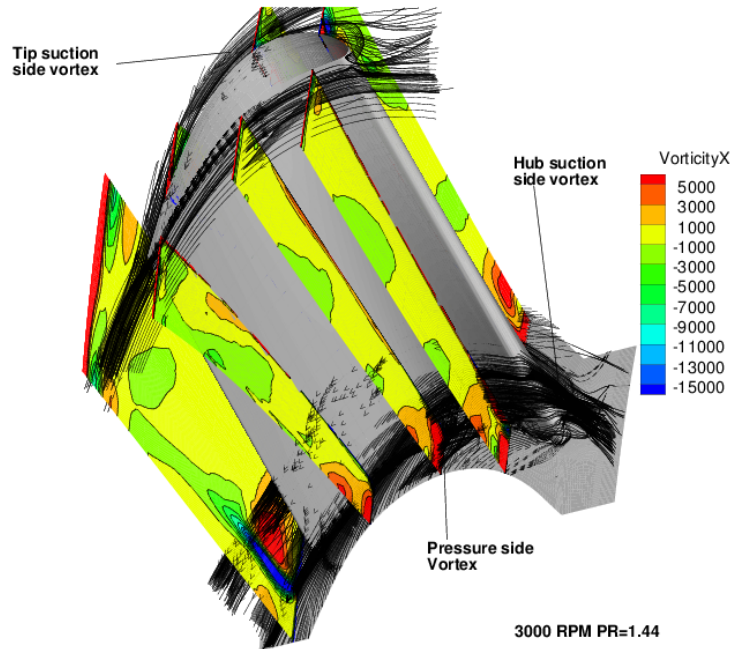


Figure 7.17: Two-stage turbine stage 2 rotor hub and tip vortices for 3000 RPM PR=1.44 case viewing the pressure side.

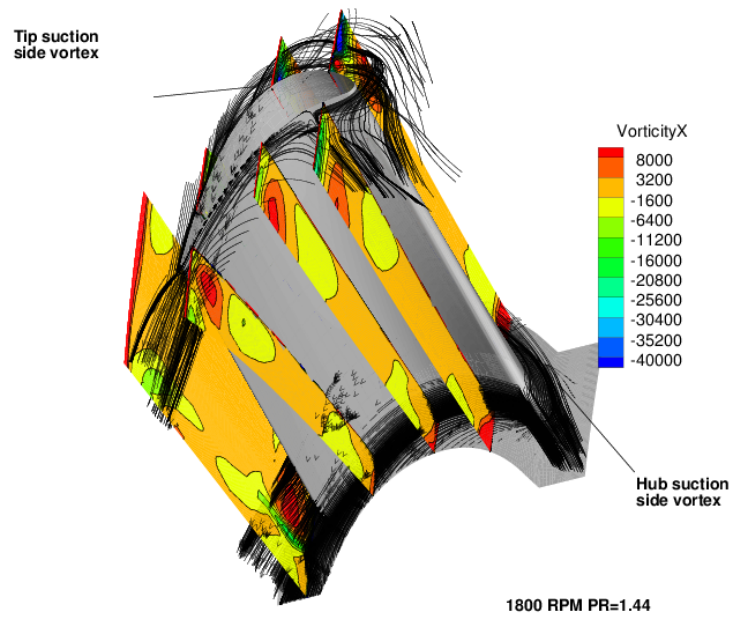


Figure 7.18: Two-stage turbine stage 2 rotor hub and tip vortices for 1800 RPM PR=1.44 case viewing the pressure side.

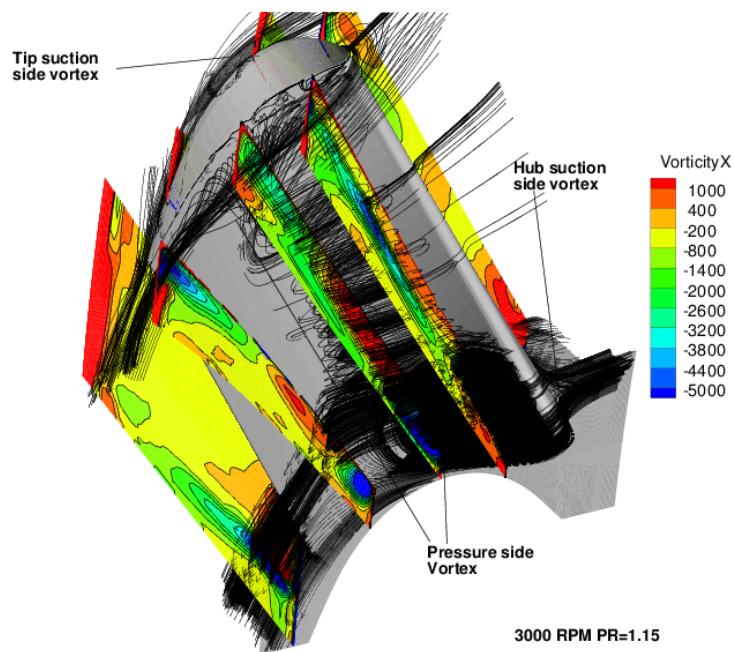


Figure 7.19: Two-stage turbine stage 2 rotor hub and tip vortices for 3000 RPM PR=1.15 case viewing the pressure side.

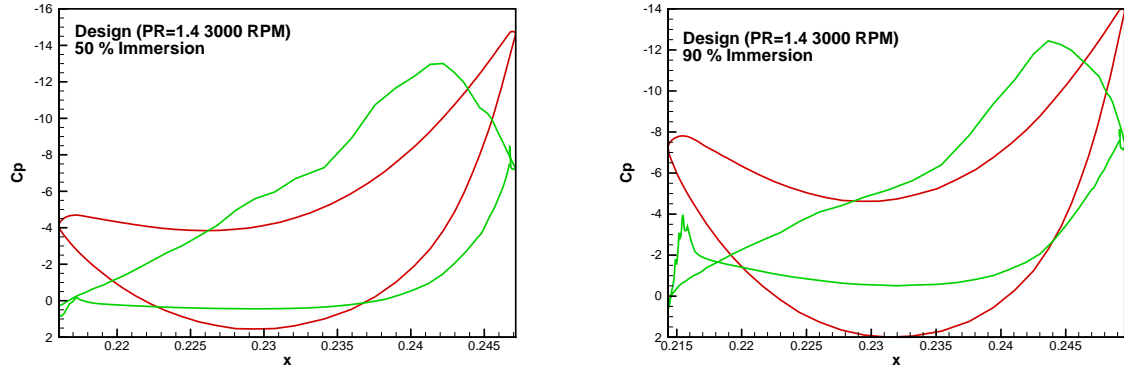


Figure 7.20: Blade loading chart (C_p versus axial position) for the design case (3000 RPM speed and PR=1.4) for 50% immersion (left) and 90% immersion (right).

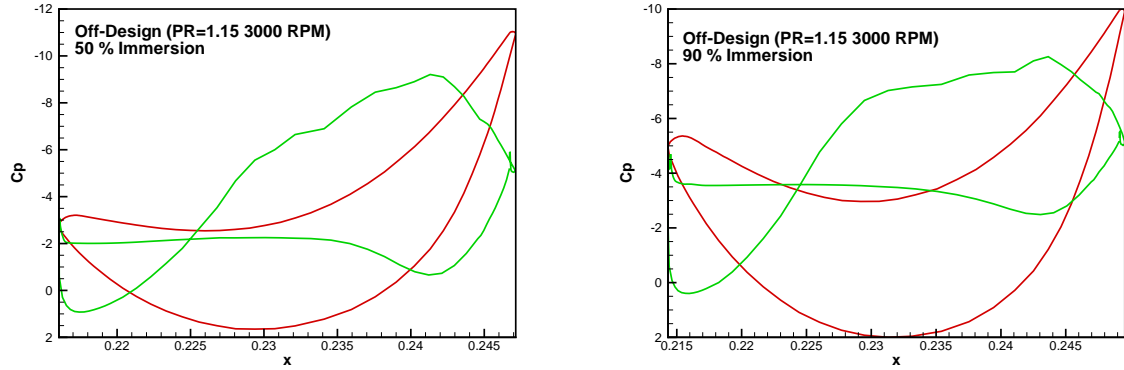


Figure 7.21: Blade loading chart (C_p versus axial position) for the off-design case (3000 RPM speed and PR=1.15) for 50% immersion (left) and 90% immersion (right).

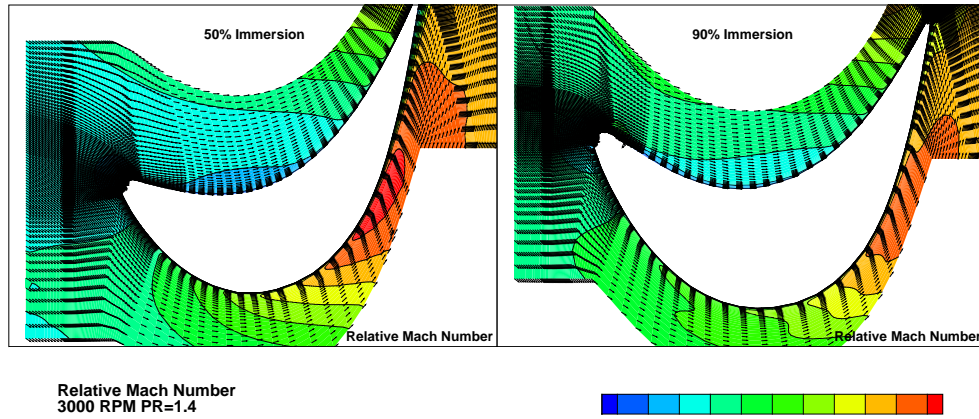


Figure 7.22: Velocity vectors and contours of relative Mach number for the design case (3000 RPM speed and PR=1.4) for 50% immersion (left) and 90% immersion (right).

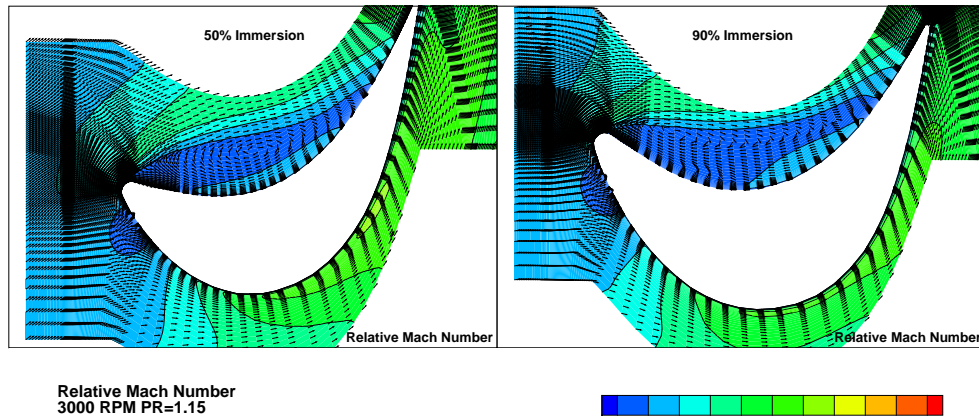


Figure 7.23: Velocity vectors and contours of relative Mach number for the off-design case (3000 RPM speed and PR=1.15) for 50% immersion (left) and 90% immersion (right).

7.2.3 Comparison to Experimental Data

Using a 3000 RPM case with a pressure ratio (PR) of 1.44 experimental results were compared to a RANS as well as a URANS analysis conducted at similar operating conditions.

Figures 7.24 to 7.30 show various parameters as a function of normalized span at stations 3, 4 and 5. The data compared includes experimental circumferentially averaged values, circumferentially averaged RANS results and finally time averaged and circumferentially averaged URANS results. Although the CFD uses inlet pressure and exit pressures that are obtained from rakes at the inlet and exit position, the inter-stage pressures are not predicted very well, with discrepancies on the order of up to 2.5 kPa in some instances. The largest difference is observed in the static pressures. Figure 7.25 shows the total pressure values, and it can be seen that although at stations 4 and 5 the discrepancy is significant, the ratio of pressure from stations 3 to 4 and 5 is similar to what is measured experimentally. It can also be observed that the URANS gives predictions that are slightly closer to the measured data, however URANS does not give an advantage when it comes to pressure predictions.

In the case of velocities, as shown in figures 7.27 and 7.28, the CFD data is much better when compared to experimentally measured absolute and relative velocities. The maximum absolute velocity is on the order of Mach 0.45, the velocities profiles show prominent 'humps' or increases in Mach number in the near-wall locations, namely at 10% immersion as well as 90% immersion, this however is not observed in the experimental data. These velocity increases are also manifested in both the absolute velocities flow angles α and relative velocity flow angles β , both shown in figures 7.29 and 7.30 respectively. The increase in flow angles, suggest flow deflection or mixing in the near wall regions due to, near wall vortices or general interaction. The near wall interactions can be observed in figure 7.35, where in both CFD (RANS and URANS) and experimental data, the wake structure at the stator exit (station 4) is clearly observed. The numerical simulations show a much thinner wake that enlarges at the near wall regions as the wake interacts with the near wall vortices. The experimental data show much larger wakes and a smaller resultant becalmed region. The near wall change in relative flow angle can clearly seen in the near-hub region of figure 7.34, as expected the near wall interactions result in high total pressure loss, especially near the hub, as shown in figures 7.31 and 7.32. The total pressure loss

coefficient is defined for the stator row as (3,4 and 5 are indices for stations 3, 4 and 5):

$$\zeta_s = \frac{P_3 - P_4}{P_3 - p_4} \quad (7.5)$$

and for the rotor low, the total pressure loss coefficient is defined as:

$$\zeta_r = \frac{P_{r4} - P_{r5}}{P_{r4} - p_5} \quad (7.6)$$

Experimentally measured loss is shown to be about half of the losses that are predicted using both steady and unsteady analyses of the turbine which contradicts the wake structures observed in the contours plots as shown in figures 7.33 to 7.35. The control of design conditions and off-design conditions were achieved by changing both the machine shaft speed, which largely results in different a flow incidence at each blade row. An off-design condition can also be achieved by changing the pressure drop and subsequently massflow rate across the machine. Experimentally these changes were achieved by using the dynamometer in order to change the shaft speed or via controlling the blower or compressor unit at the exit to control the pressure drop and subsequently increase or decrease the massflow as needed.

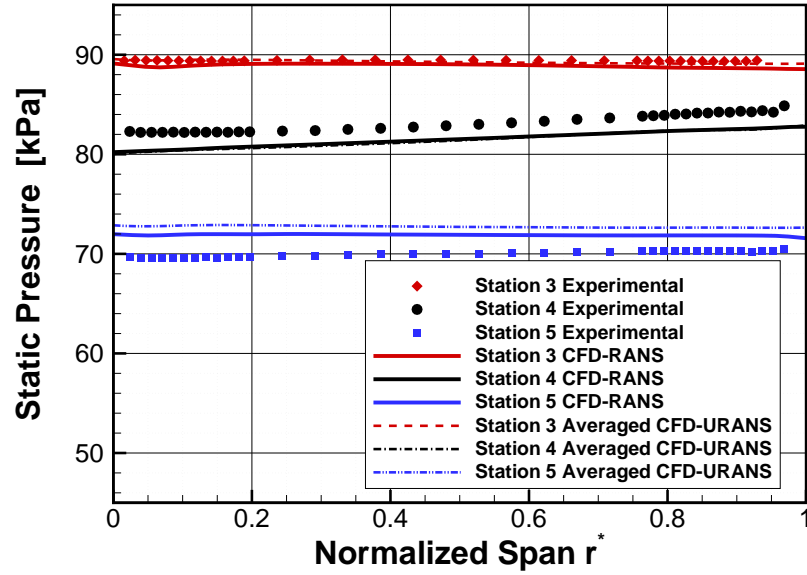


Figure 7.24: Two-stage turbine spanwise averaged static pressure at PR=1.44 and 3000 RPM.

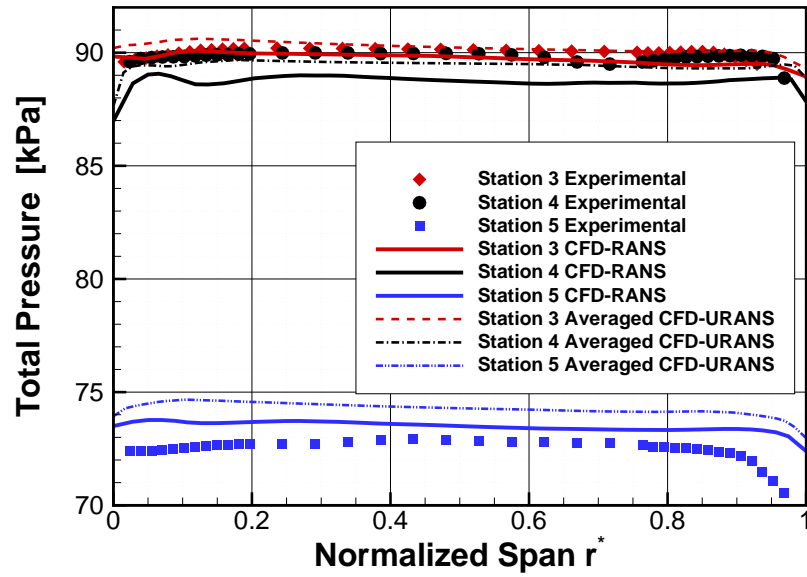


Figure 7.25: Two-stage turbine spanwise averaged total pressure at PR=1.44 and 3000 RPM.

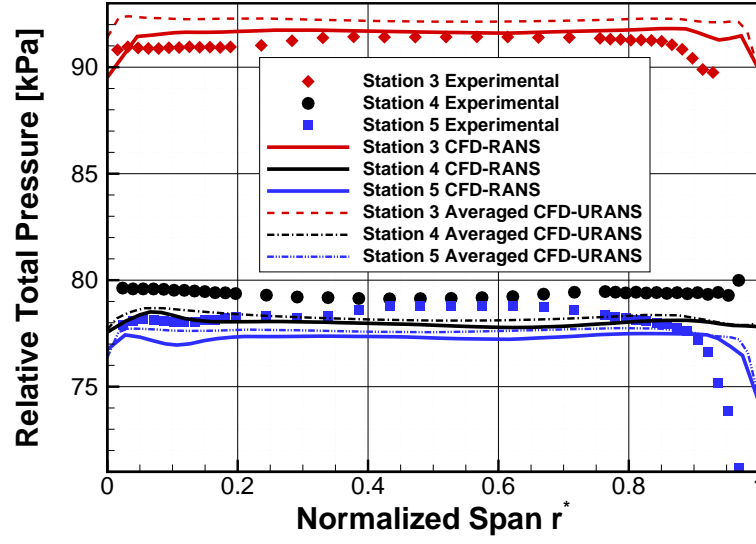


Figure 7.26: Two-stage turbine spanwise averaged relative total pressure at PR=1.44 and 3000 RPM.

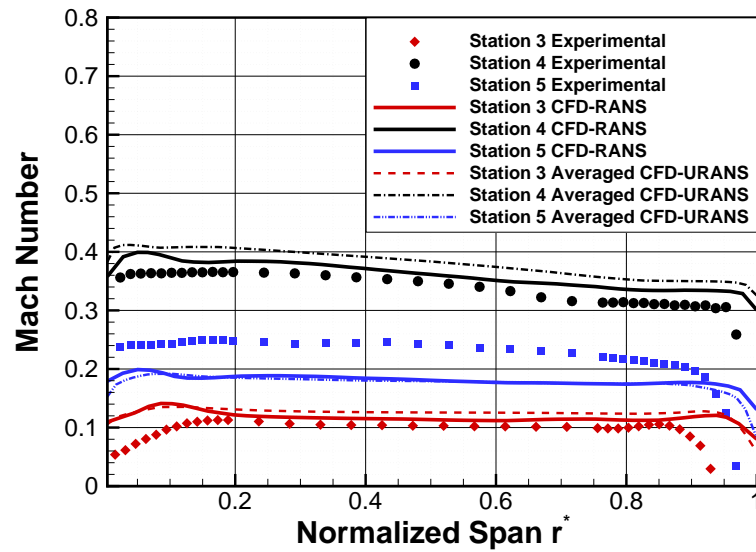


Figure 7.27: Two-stage turbine spanwise averaged Mach number at PR=1.44 and 3000 RPM.

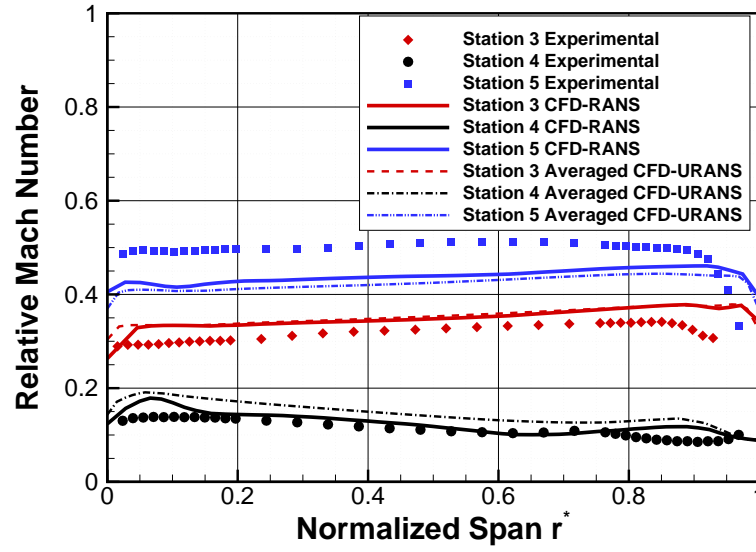


Figure 7.28: Two-stage turbine spanwise averaged relative Mach number at PR=1.44 and 3000 RPM.

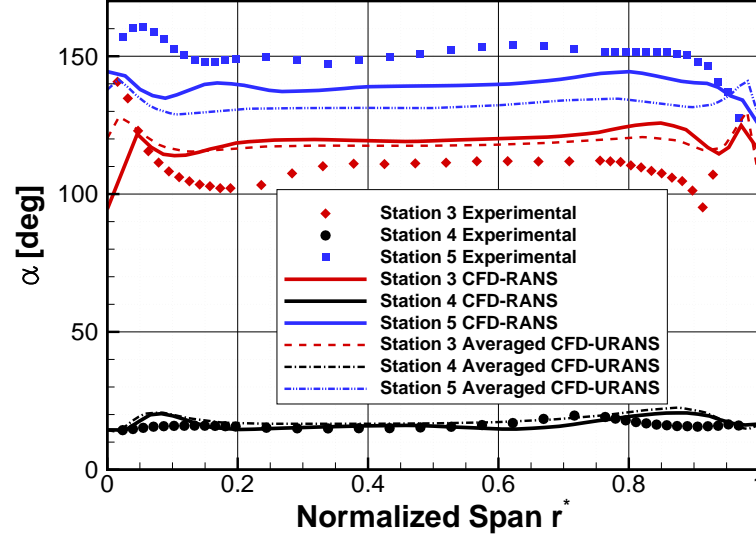


Figure 7.29: Two-stage turbine spanwise averaged absolute flow angle α at PR=1.44 and 3000 RPM.

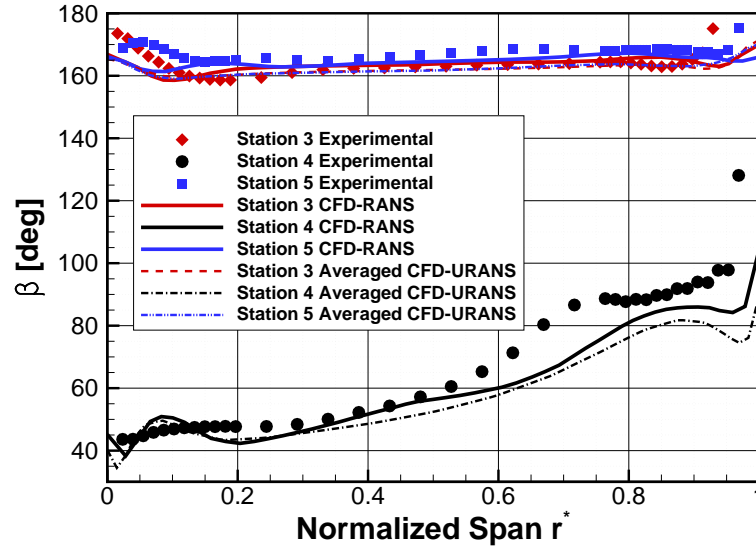


Figure 7.30: Two-stage turbine spanwise averaged relative flow angle β at PR=1.44 and 3000 RPM.

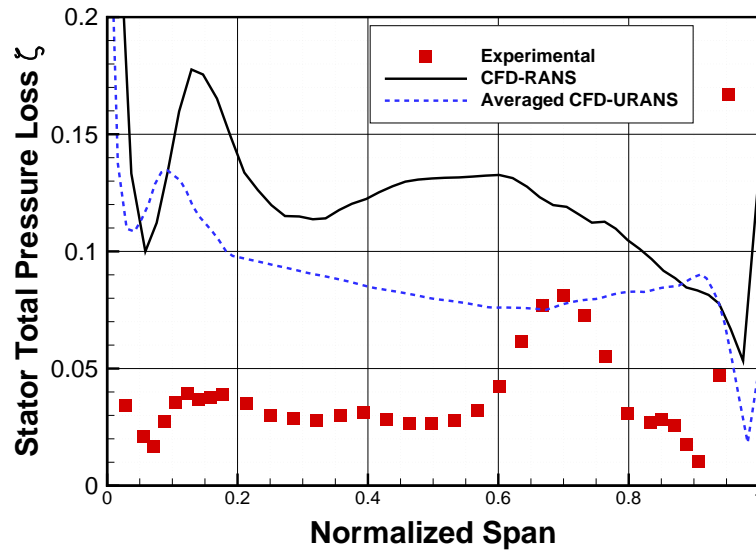


Figure 7.31: Two-stage turbine stator 2 total pressure loss ζ versus span at PR=1.44 and 3000 RPM.

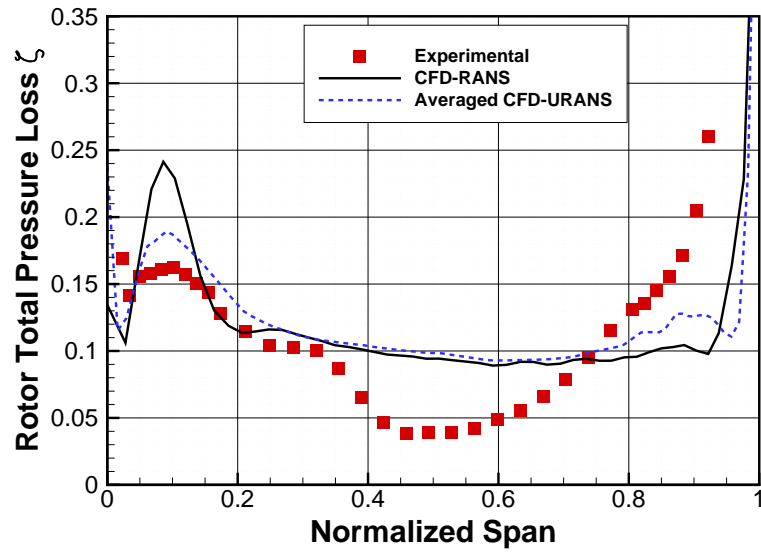


Figure 7.32: Two-stage turbine rotor 2 total pressure loss ζ versus span at PR=1.44 and 3000 RPM.

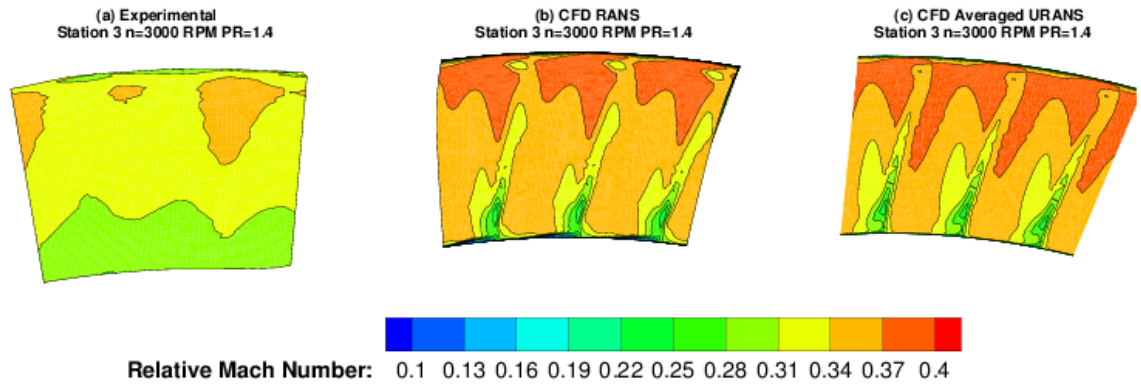


Figure 7.33: Two-stage turbine contours of relative Mach number at station 3 for (a) experimental (b) RANS and (c) averaged URANS at PR=1.44.

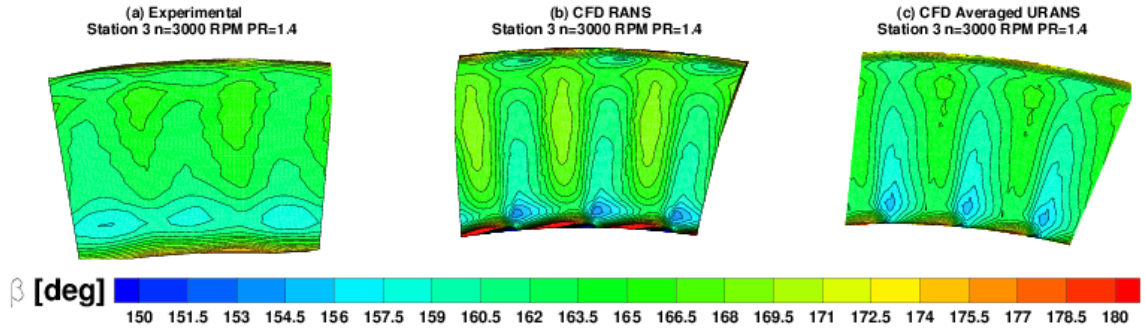


Figure 7.34: Two-stage turbine contours of relative flow angle β at station 3 for (a) experimental (b) RANS and (c) averaged URANS at PR=1.44.

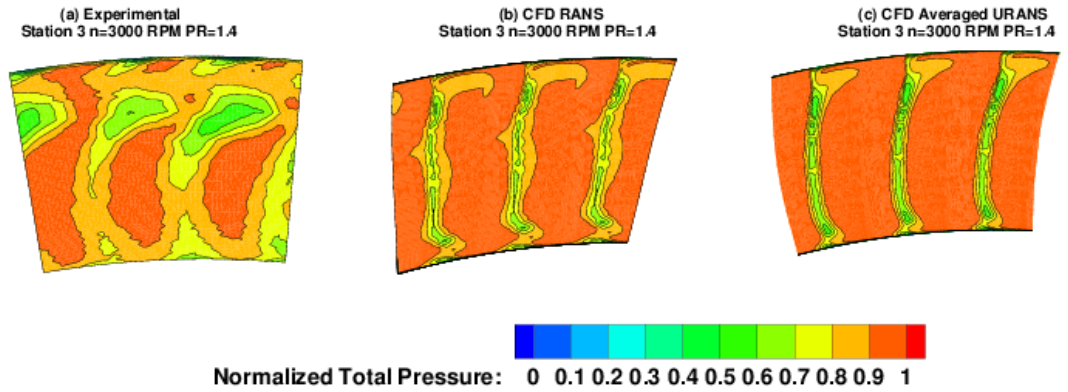


Figure 7.35: Two-stage turbine contours of normalized total pressure at station 4 for (a) experimental (b) RANS and (c) averaged URANS at PR=1.44.

7.2.4 Tip Leakage Effects

One important consideration in turbomachinery design and performance is the tip clearance between rotor and casing in unshrouded rotor blades. In this particular configuration the rotors are fully shrouded and thus tip leakage is not a consideration, however it may be a valuable exercise to observe the potential effects of tip leakage on this rotor design.

In order to do so the rotor domains were remeshed to include a tip region. The tip region uses 11 node points and the first node is sized to achieve a Y^+ value that is close

to 1. The tip region was assumed to be about 1% to 2% of the blade height which in this case translates to 0.635-1.27 millimeters, the mesh of the first stage rotor is shown in figure 7.36. The new mesh with tip clearance regions in the rotor rows was run at the same design points used with the shrouded mesh and a comparison was made from a performance and flow visualization point of view. Two cases were run, a 1800 RPM case as well as a 3000 RPM case at maximum pressure ratio of approximately 1.4. Figure 7.37 shows the overall total-to-static efficiency versus U/C_o for various configurations including shrouded CFD (RANS), shrouded experimental, 1% rotor tip and finally 2% rotor tip. As shown the RANS solver shows a large discrepancy between RANS predicted as and experimentally measured efficiency values. At the highest efficiency point the discrepancy is about 5 efficiency points.

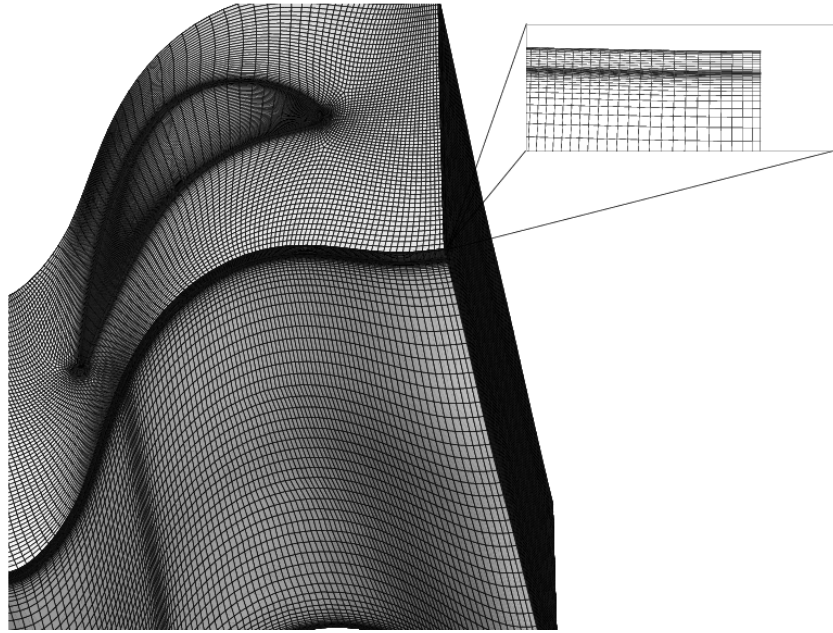


Figure 7.36: Two-stage turbine rotor blade mesh using a tip clearance region.

Figures 7.38 and 7.39 show flow stream of the leakage area for the 1800 RPM and 3000 RPM cases respectively. As observed earlier the 1800 RPM which in this case is considered

an off-design case shows a high negative incidence in the tip region, which results in two large corner vortices occurring between both the pressure and suction surfaces and the tip leakage area. The two vortices exist in the high shaft speed 3000 RPM case but have much lower intensity, as also shown in the axial vorticity contour slices shown for both cases in figures 7.40 and 7.41. The patterns observed are consistent with those discussed by the authors in [94]. Both vorticities at the suction and pressure surfaces are positive since the tip leakage flow would progress from the pressure surface to the suction surface. At the 1800 RPM case the tip leakage is much higher due to the high negative flow incidence at the tip region when compared to the 3000 RPM operating point.

Generally from an experimental point of view it is difficult to determine exactly the tip clearance size at various running conditions. Various experimental studies have been performed on tip clearance leakage and their interactions with the blade passage including Dambach et al [95] and [96]. The new mesh with tip clearance regions in the rotor rows was run at the same operating points as used with the shrouded mesh, and a comparisons was made from a performance and flow visualization point of view. Specifically two cases were run, a 1800 RPM case as well as a 3000 RPM case at a maximum pressure ratio of approximately 1.4 for both. Figure 7.37 shows the overall total-to-static efficiency η_{ts} versus U/C_o for various configurations including shrouded CFD (RANS), Experimental, a 1% rotor tip and finally a 2% rotor tip. The dimensionless parameter U/C_o is defined as:

$$U = \frac{\pi n}{60} (r_{hub} + r_{tip}) \quad (7.7)$$

$$C_o = \sqrt{\frac{2\Delta H}{n_{stages}}} \quad (7.8)$$

$$\frac{U}{C_o} = \frac{\frac{\pi n}{60} (r_{hub} + r_{tip})}{\sqrt{\frac{2\Delta H}{n_{stages}}}} \quad (7.9)$$

As shown, the RANS solver in general shows a large discrepancy between RANS predicted and experimentally determined machine efficiency. At the highest efficiency point ($U/C_o \approx 0.7$) the discrepancy is about 2.5 to 3 efficiency points. The possible reasons for

such a large discrepancy will be discussed in the discussion section of this thesis, however in general the use of CFD as an a priori predictor of machine efficiency is quite risky. A general statement about the discrepancy could be attributed to model deficiencies in predicted secondary flows and secondary flow mixing effects as well as properly accounting for transition. CFD codes are found to mix out any secondary effects rather rapidly when compared to experimentally observed secondary flow tendencies. This is shown clearly in figures 7.33 to 7.35 where the wakes are clearly shown to much smaller in spanwise thickness and locations comparable to experimental data. Nonetheless the use of CFD as means to compare one operating point to the other provides useful information, and when comparing relative operating points, predicted by CFD and measured by experiments shows good correlations.

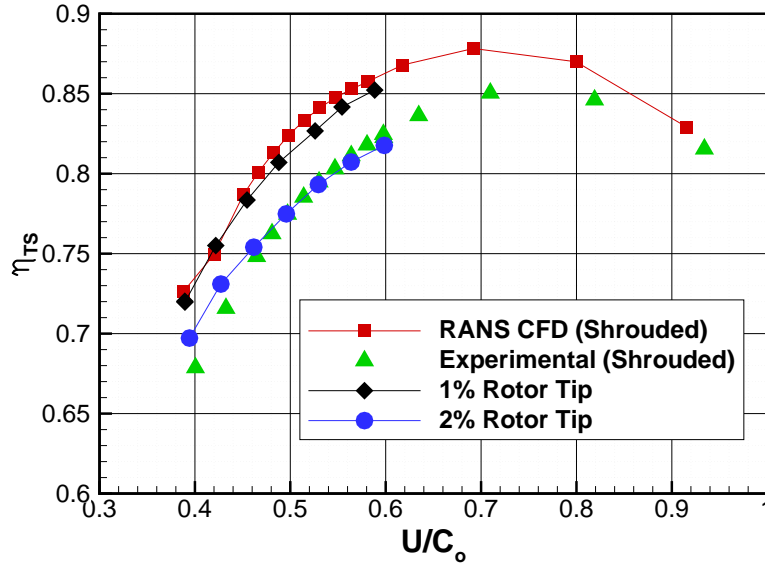


Figure 7.37: Two stage turbine total-to-static efficiency versus U/C_o .

The use of 1% and 2% tips in the two rotor rows causes significant effects on the efficiency especially at points that are closer to design points. At low U/C_o values the difference between in efficiency between the shrouded, 1% tip and 2% tip configurations is small, however this difference increases as the operating point is advanced. The mecha-

nism for the increased loss is largely due to interactions between the leakage flows and the tip vortices, this mixing or interaction between the leakage loss and the tip vortices will generate greater losses, this effect will be exacerbated at higher U/C_o values due to the higher pressure ratios which in turn will increase the leakage flows.

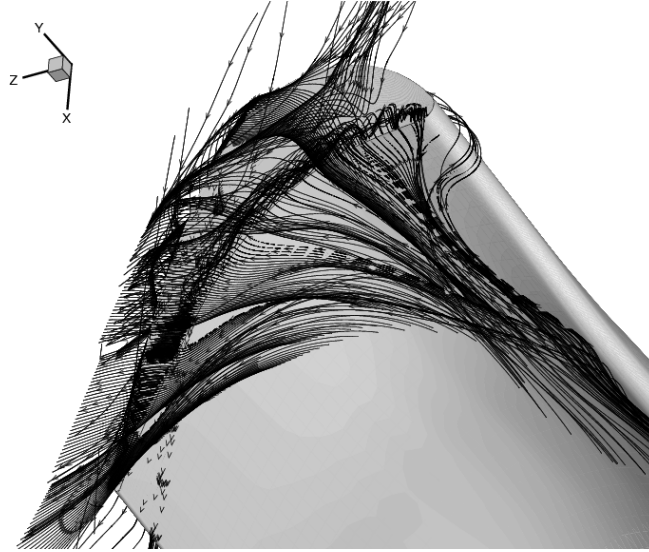


Figure 7.38: Tip leakage streamlines at 1800 RPM.

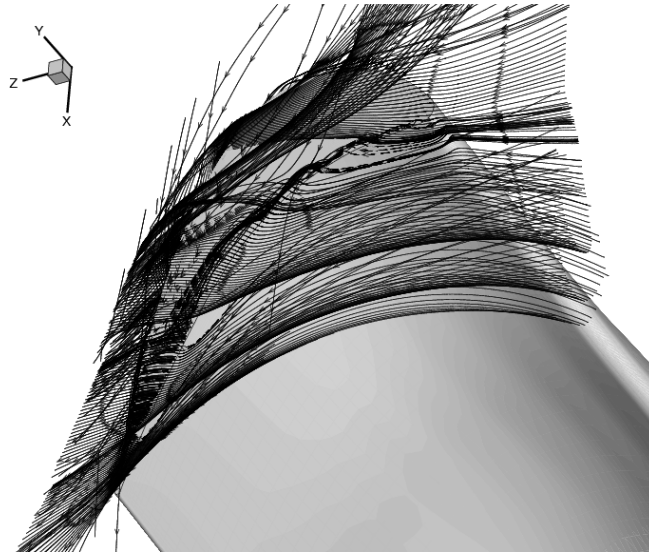


Figure 7.39: Tip leakage streamlines at 3000 RPM.

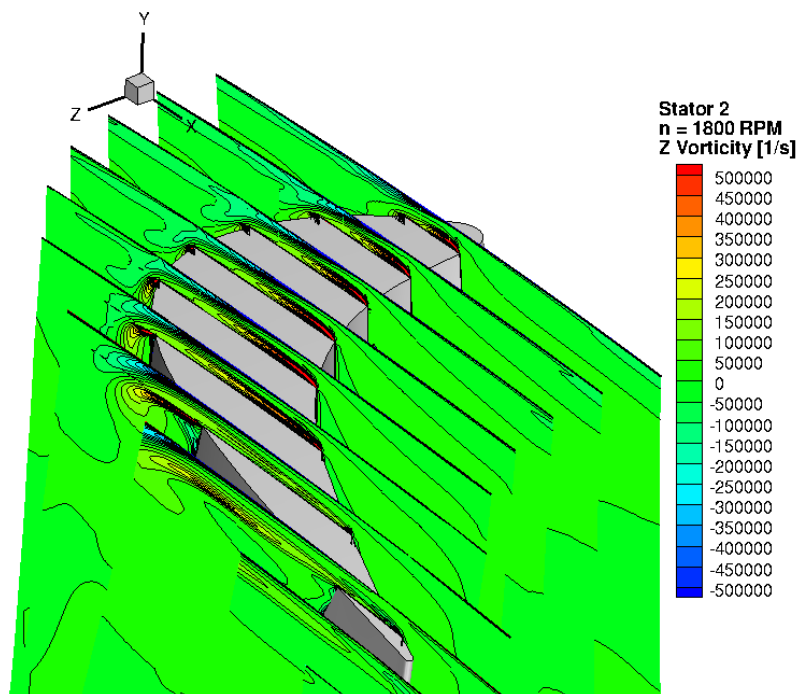


Figure 7.40: Tip leakage vorticity contours for the 1800 RPM case.

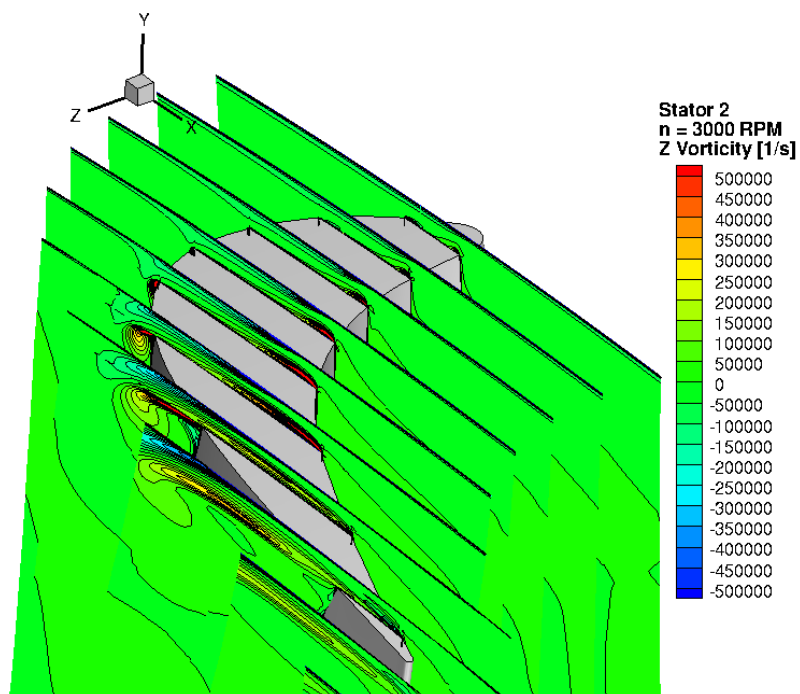


Figure 7.41: Tip Leakage vorticity contours for the 3000 RPM case.

8. THREE STAGE TURBINE STUDY*

In this chapter* a three stage turbine design is examined that uses both cylindrical as well as bowed or 3D blades for both the stator and rotor rows. The effects of bowing are examined and compared to experimental data for both designs, moreover a look at the unsteady effects is observed via the use of URANS. The cylindrical blades and the 3D bowed blades were the first designs tested at the TPFL in the late 1990s, where each design's performance characteristics were examined.

8.1 Blade Designs and Mesh

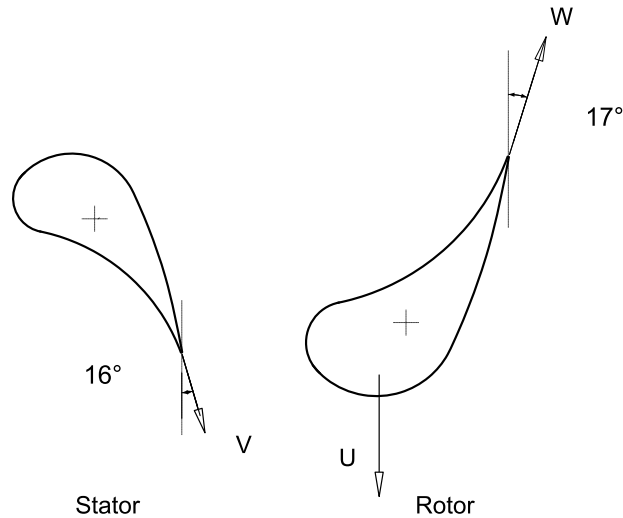


Figure 8.1: Cylindrical blade series, showing both stator and rotor profiles and blade exit metal angles.

The 2D profiles for the cylindrical blade series are shown in figure 8.1, as shown the

*Part of the data reported in this chapter is reprinted with permission from "Experimental and Numerical Investigations of Aerodynamic Behavior of a Three-Stage High-Pressure Turbine at Different Operation Conditions " by S. Abdelfattah and M.T. Schobeiri, 2012. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 226, 1535-1549. Copyright 2012, by the Institution of Mechanical Engineers.

stators have an exit metal angle of 16 degrees while the rotor exit metal angle is 17 degrees, the blade heights for both stator and rotor stages are 2.5 inches or 63.5 mm. Figure 8.2 shows the 3D bowed blade series rotor profiles as well as the bowing and sweep angles particularly at hub and tip regions. The blade height for the 3D bowed blades is the same as the cylindrical blades with a blade height of 63.5 mm (2.5 inches).

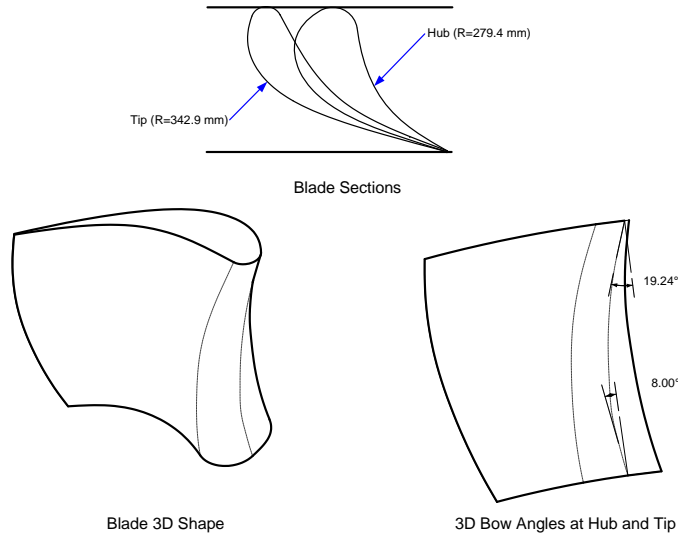


Figure 8.2: 3D bowed blade rotor profile and bowing angles.

Using the discretizations and methodology employed in 7.1.4 a numerical discretization was used for both cylindrical and 3D bowed blade designs. Table 8.1 lists various parameters related to both designs. The blade numbering is of note, which not an issue for the RANS modeling due to the use of the mixing plane method, but for the unsteady cases due to the need for a 1:1 interface area matching the blade numbering was set to 50 for all rows. The impact of this change is discussed extensively in the discussion section of this chapter. The hub and tip seals were not considered in the numerical model in order to simplify the model, moreover the both stator and rotor rows were shrouded and thus a no considerations for a hub leakage (stator) or a tip leakage (rotor) were needed. An annular geometry was created and imported into ICEM-CFD. Using a blocking technique for each

half stage and seal geometry a structured hexahedral mesh was created. The mesh created is shown in figures 8.3 for the cylindrical blades and figure 8.4 for the 3D bowed blade design. The created stator mesh has around 600,000 hexahedral elements whereas the rotor elements have about 1.8 million hexahedral elements. The hub and tip seal domains each are in the order of 200,000 elements. The total element number for the three stages is approximately 8 million elements. The wall boundary layer discretization used 15 points for the stator domains and 20 points for the rotor walls.

The boundary conditions used were obtained from inlet and outlet measurement rakes used in the facility. The inlet rakes are placed directly after the inlet heater, and the outlet rakes are placed directly after the third rotor stage. The rakes include total and static pressure probes as well as J-type thermocouples. Referring to figure 8.5, numerical domain inlet and outlet points are measurement station 1 (St1) and station 7 (St7) respectively.

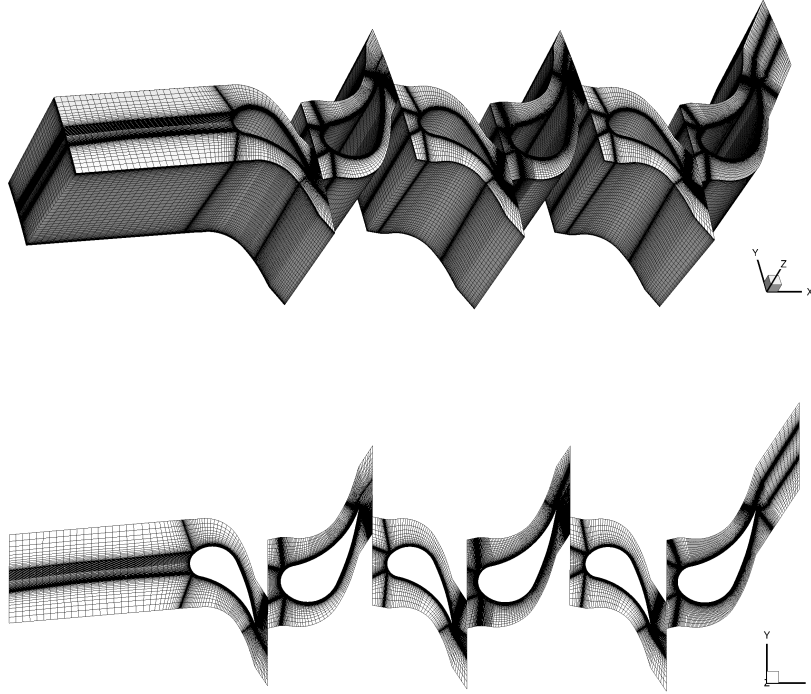


Figure 8.3: Cylindrical blade series mesh.

Table 8.1: 3-stage cylindrical and bowed blade turbine specifications.

Item	Specifications
Stage Number	$N = 3$
Tip Diameter	$D_t = 685.8$ mm
Hub Diameter	$D_h = 558.8$ mm
Blade Height	$B_h = 63.5$ mm
Power	$P = 80.0\text{-}110.0$ kW
Mass Flow	$\dot{m} = 3.728$ kg/s
Speed Range	$n = 1800\text{-}2800$ rpm
Inlet Pressure	$p_{in} = 101.356$ kPa
Exit Pressure	$p_{ex} = 71.708$ kPa
Stator Blade Number	S1=58 S2=52 S3=56
Rotor Blade Number	R1=46 R2=40 R3=44

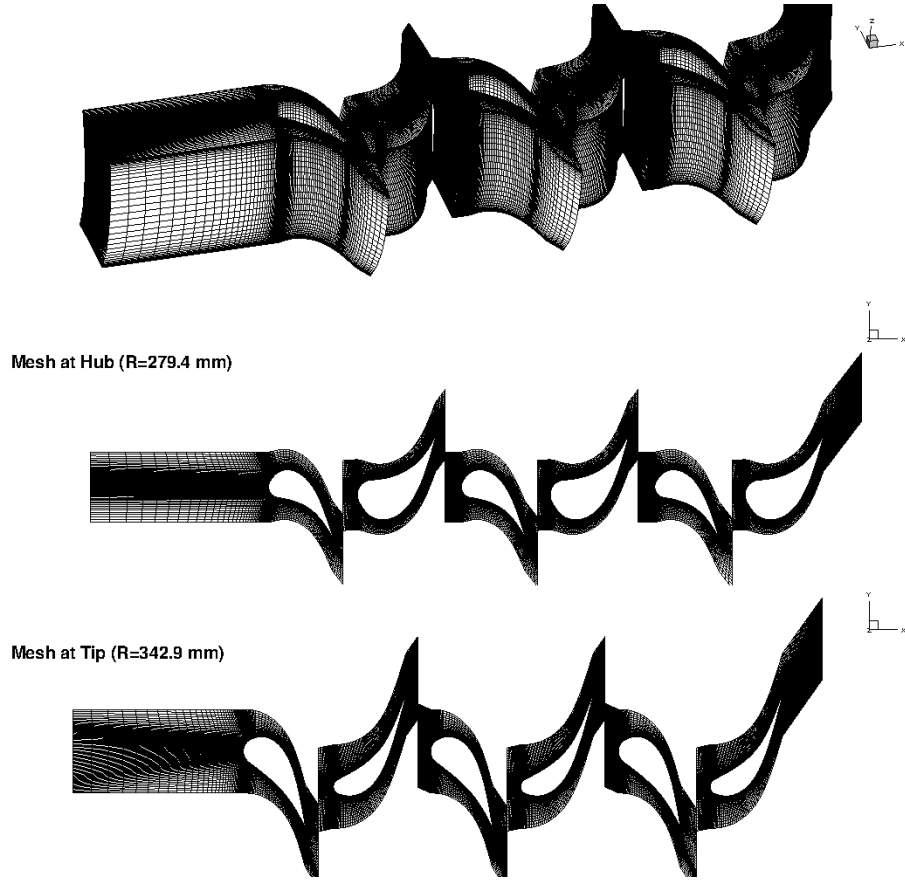


Figure 8.4: 3D bowed blade series mesh.

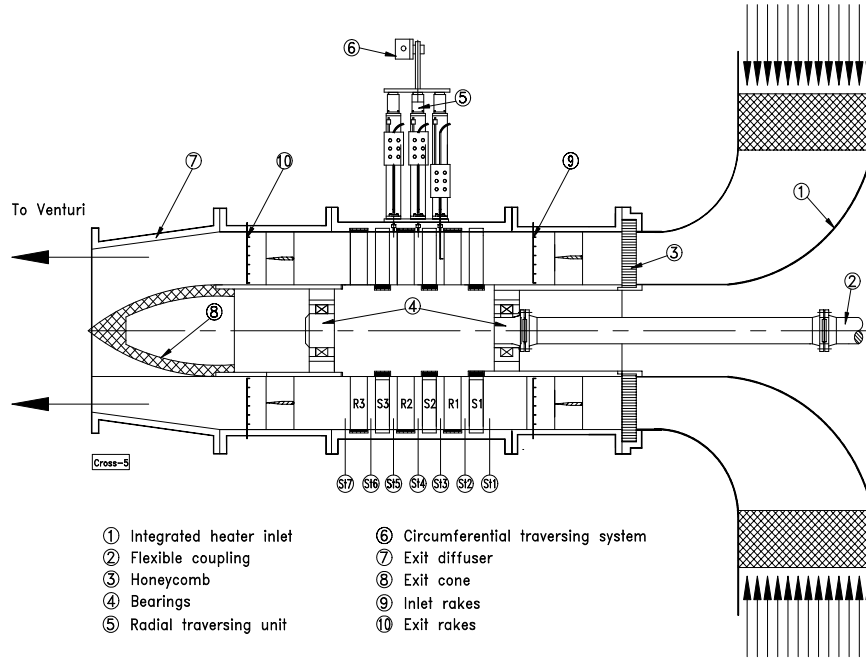


Figure 8.5: TPFL three stage configuration cross-section, obtained from [91].

8.2 Numerical Simulation

Each half stage was defined as a domain within the ANSYS-CFX workspace, and specific interfaces are defined for the seals' entrance and exit points for both the rotor and

Table 8.2: Boundary conditions for 1800 RPM and 2600 RPM cases.

Boundary Condition	1800 rpm	2600 rpm
Reference Pressure [kPa]	101.261	100.975
Total Pressure Inlet [kPa]	100.496	100.271
Static Pressure Outlet [kPa]	71.654	72.021
Total Temperature Inlet [K]	323.48	324.0
Total Temperature Outlet [K]	298.3	297.05

stator stages. Between the stator and rotor stages a “Stage” interface is used which according to [97] performs a circumferential average of the fluxes through bands on the interface between the stator and rotor. This model allows for steady state predictions to be obtained for multi-stage machines. In addition, circumferential periodic boundary conditions are placed on the upper and lower surface of both the stator and rotor domains as well as the hub and tip seal domains. Upon application of the boundary conditions stated in Table 8.2, a steady-state computation was run until convergence was achieved. Air was selected as a medium, with assumed ideal gas behavior.

8.3 RANS Based Results

Upon convergence the numerical results were analyzed and compared to the previously obtained inter-stage experimental results, different parameters calculated with the numerical solver were compared. Convergence was deemed achieved when RMS residual values for the Reynolds Averaged Navier-Stokes (RANS) equations, and the two turbulence model transport equations reached a magnitude below 1×10^{-6} . As previously mentioned the main experimental and numerical positions in the facility that were probed are measurement stations 3 to 5 which are shown in Fig. 8.5. The majority of the experimental data used for comparison was obtained from Schobeiri et al. [91] and a consistent arithmetic averaging technique similar to the methodology used in Schobeiri et al. [91] was used to average the obtained data circumferentially. This averaging scheme was employed in both experimental and numerical data analysis.

8.3.1 Three Stage Turbine Cylindrical Blade RANS Results

In this section RANS based results for the cylindrical series blade will be compared to corresponding experimental data. Figures 8.6 and 8.7 show the static pressure at stations 3, 4 and 5 for shaft speeds of 2600 RPM and 2000 RPM respectively. The RANS solver, as observed in the 2-stage turbine case shows a significant discrepancy in pressure between measured and calculated pressure. However this discrepancy is limited to a maximum of 2kPa in most cases.

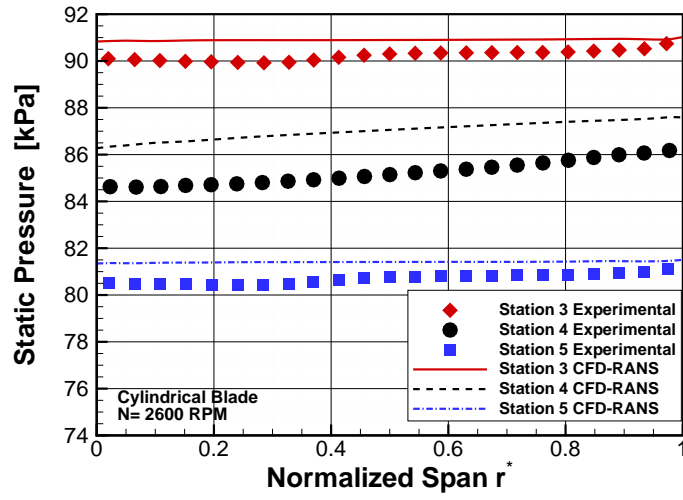


Figure 8.6: Cylindrical blade circumferentially averaged static pressure at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.

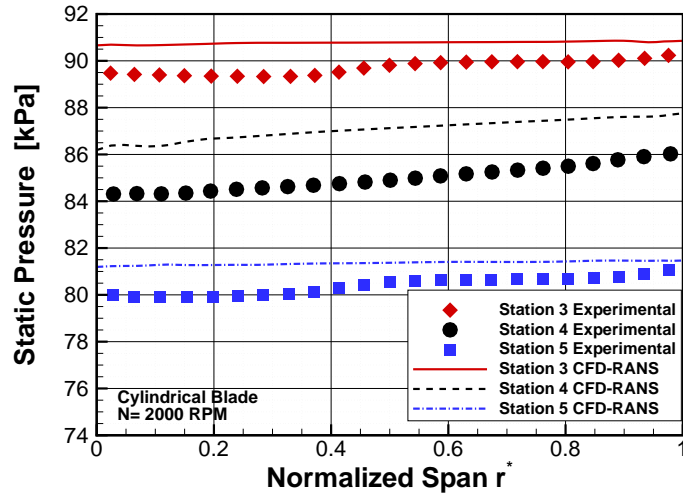


Figure 8.7: Cylindrical blade circumferentially averaged static pressure at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.

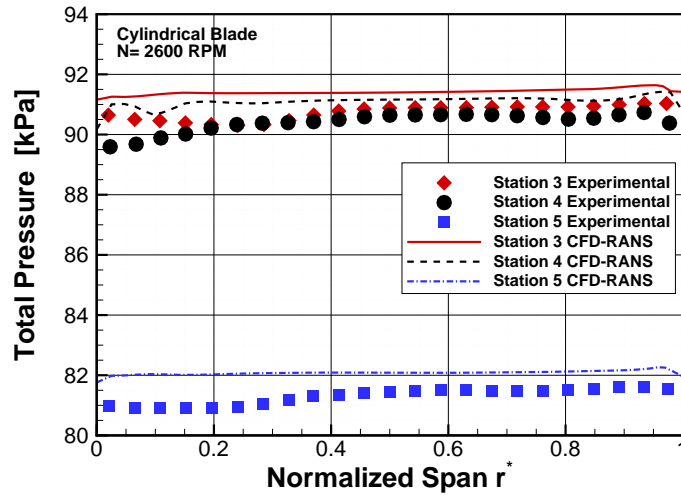


Figure 8.8: Cylindrical blade circumferentially averaged total pressure at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.

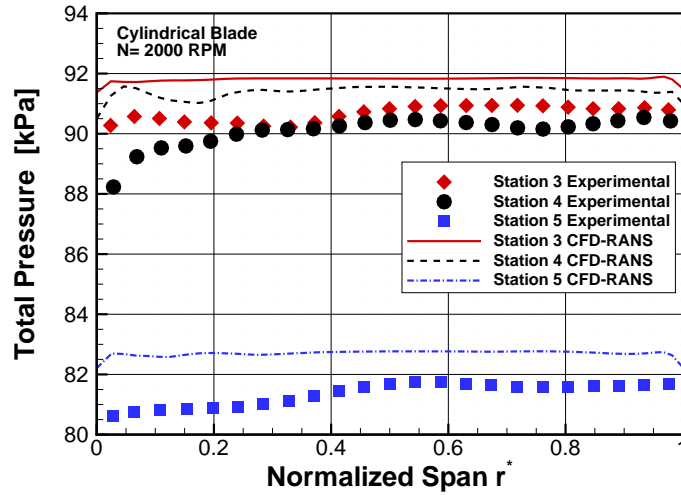


Figure 8.9: Cylindrical blade circumferentially averaged total pressure at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.

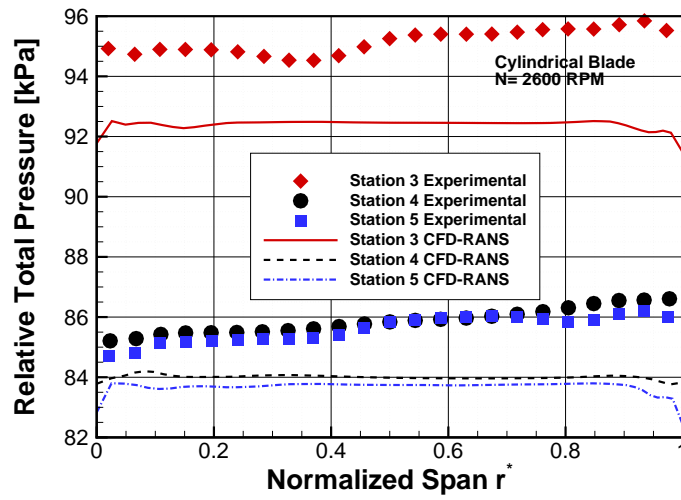


Figure 8.10: Cylindrical blade circumferentially averaged relative total pressure at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.

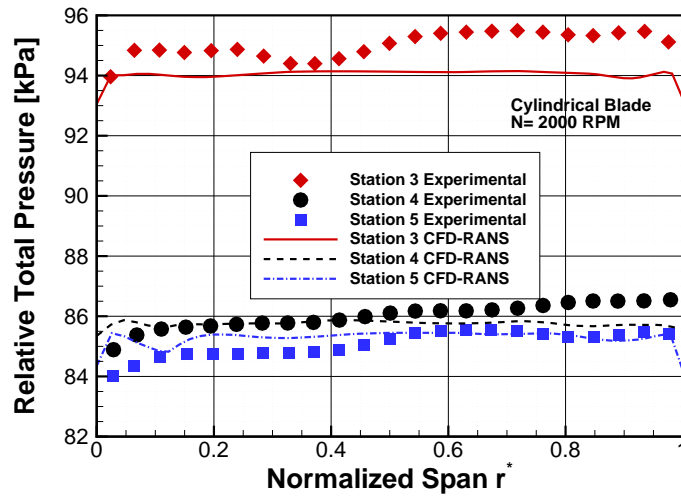


Figure 8.11: Cylindrical blade circumferentially averaged relative total pressure at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.

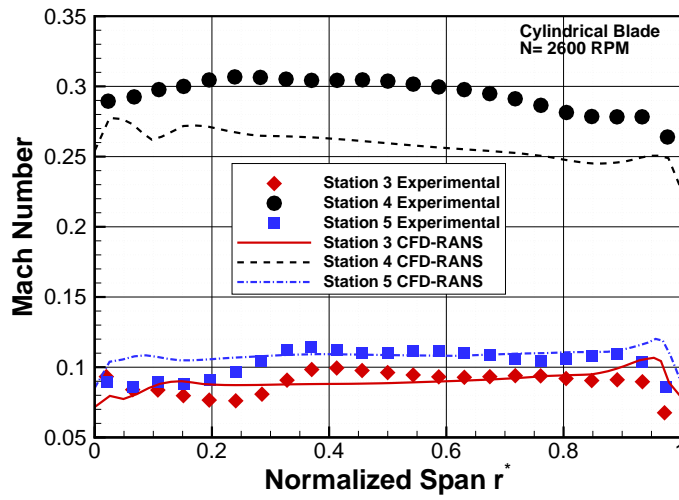


Figure 8.12: Cylindrical blade circumferentially averaged absolute Mach number at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.

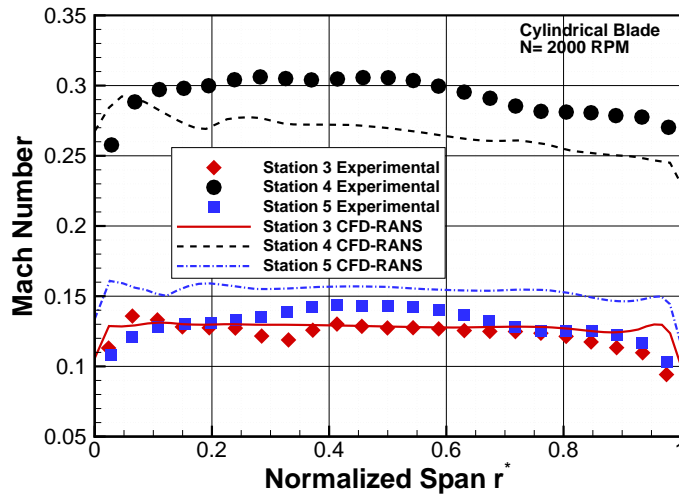


Figure 8.13: Cylindrical blade circumferentially averaged absolute Mach number at 2000 RPM for stations 3, 4 and 5 for the three-stage turbine.

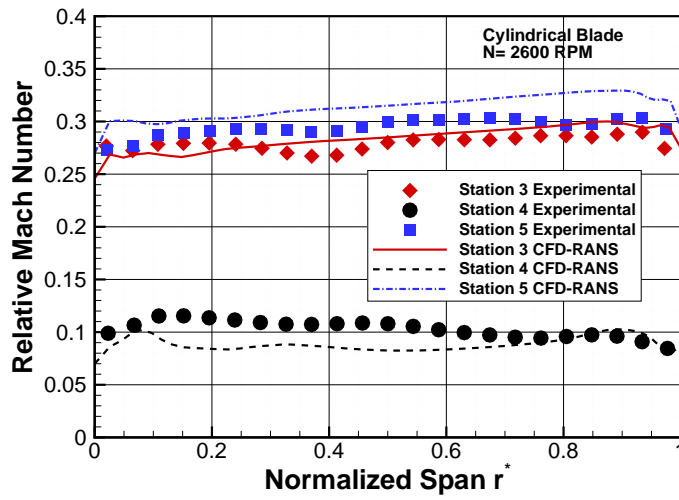


Figure 8.14: Cylindrical blade circumferentially averaged relative Mach number at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.

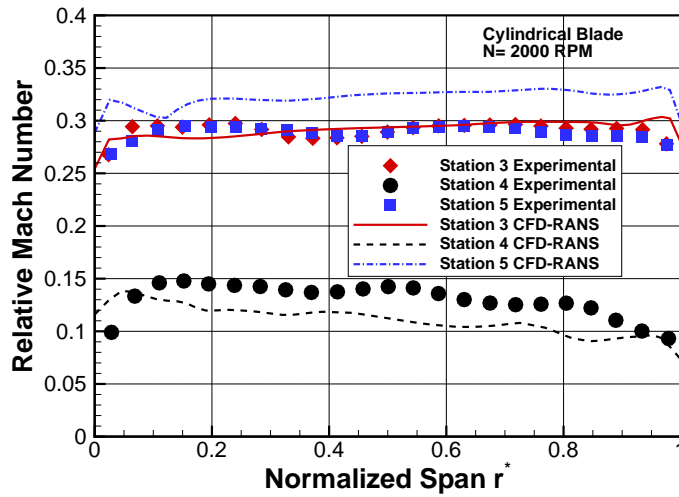


Figure 8.15: Cylindrical blade circumferentially averaged relative absolute Mach number at 2000 RPM for Stations 3, 4 and 5 for the three-stage turbine.

**Cylindrical Blade
PR=1.4 2600 RPM
Normalized Total Pressure at Station 4**

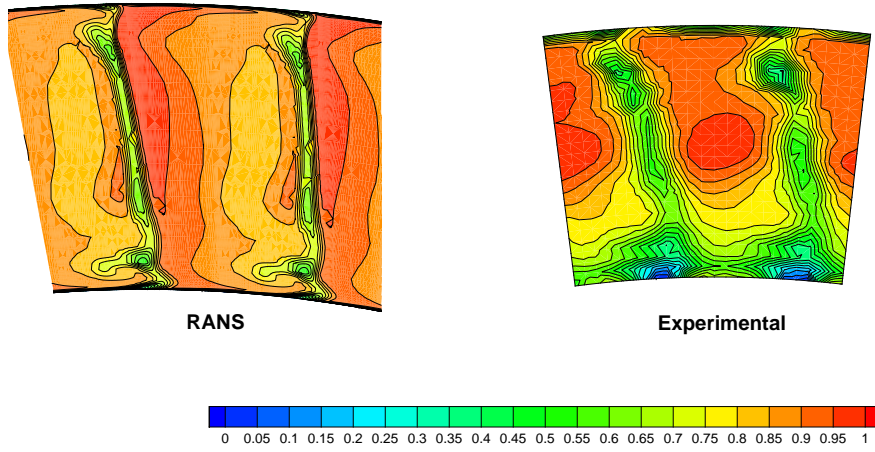


Figure 8.16: Cylindrical blade experimental and RANS normalized total pressure contour at station 4 and RPM=2600 for the three-stage turbine.

**Cylindrical Blade
PR=1.4 2000 RPM
Normalized Total Pressure at Station 4**

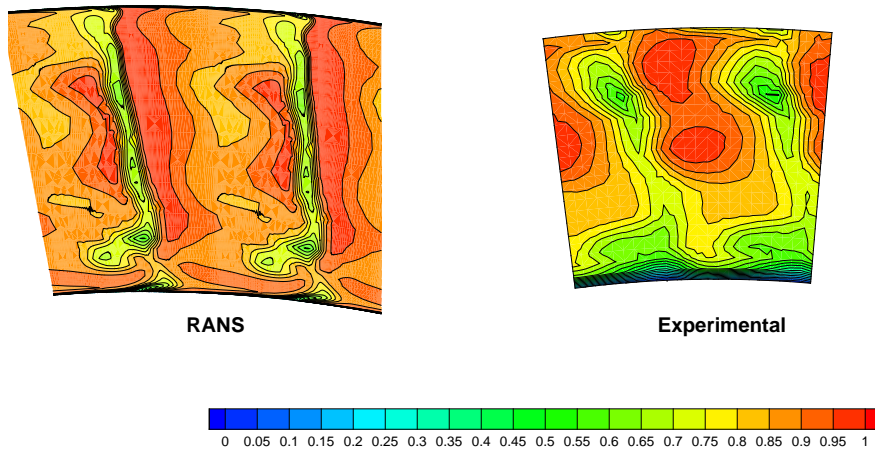


Figure 8.17: Cylindrical blade experimental and RANS normalized total pressure at station 4 and RPM=2000 for the three-stage turbine.

**Cylindrical Blade
PR=1.4 2600 RPM
Normalized Relative Total Pressure at Station 3**

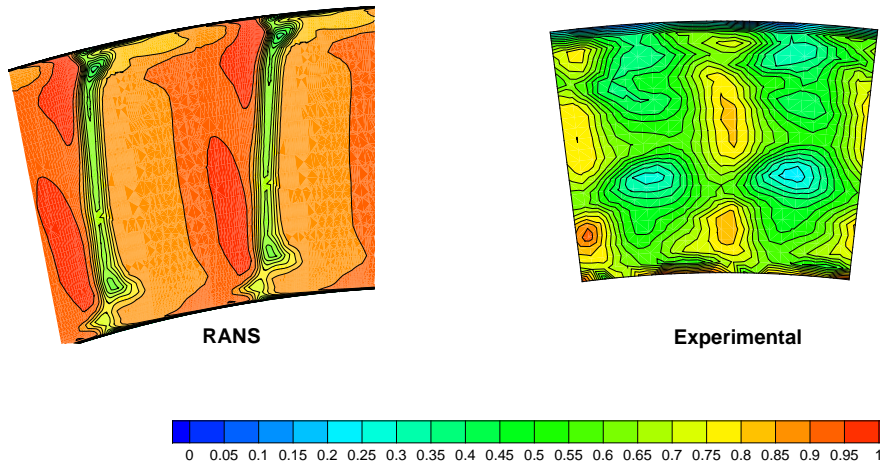


Figure 8.18: Cylindrical blade experimental and RANS normalized relative total pressure contour at station 3 and RPM=2600 for the three-stage turbine.

**Cylindrical Blade
PR=1.4 2000 RPM
Normalized Relative Total Pressure at Station 3**

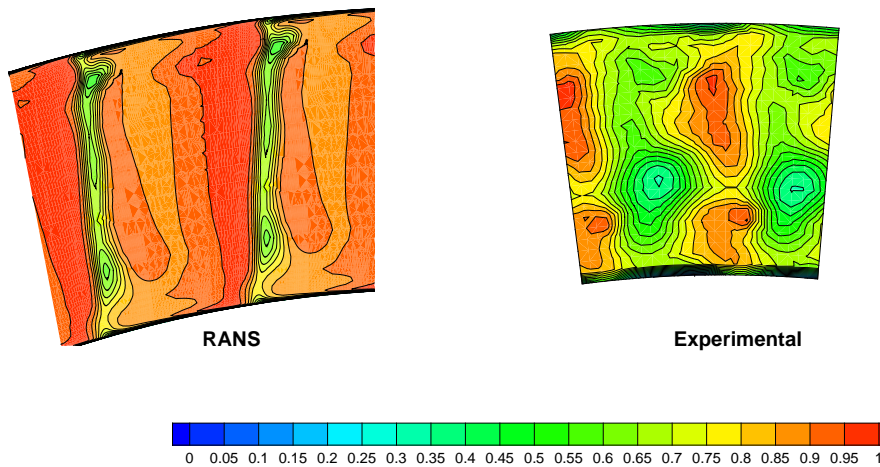


Figure 8.19: Cylindrical blade experimental and RANS normalized relative total pressure contour at station 3 and RPM=2000 for the three-stage turbine.

Cylindrical Blade
PR=1.4 2600 RPM
Mach Number at Station 4

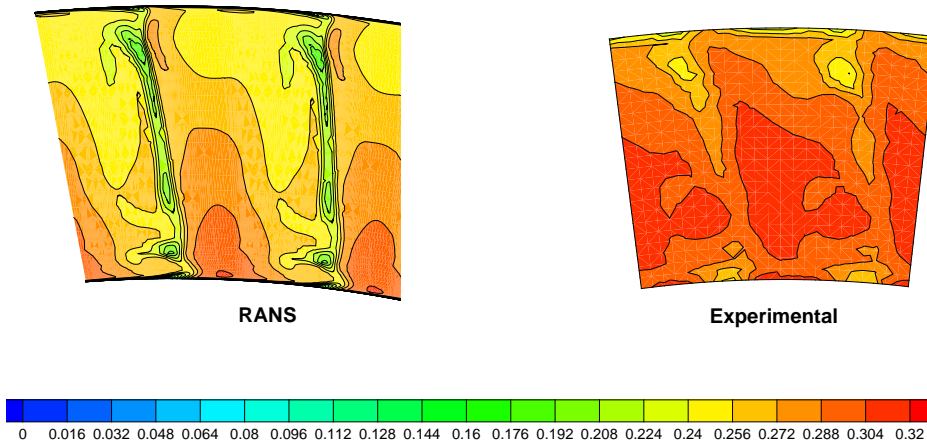


Figure 8.20: Cylindrical blade experimental and RANS Mach contours at station 4 and RPM=2600 for the three-stage turbine.

Cylindrical Blade
PR=1.4 2000 RPM
Mach Number at Station 4

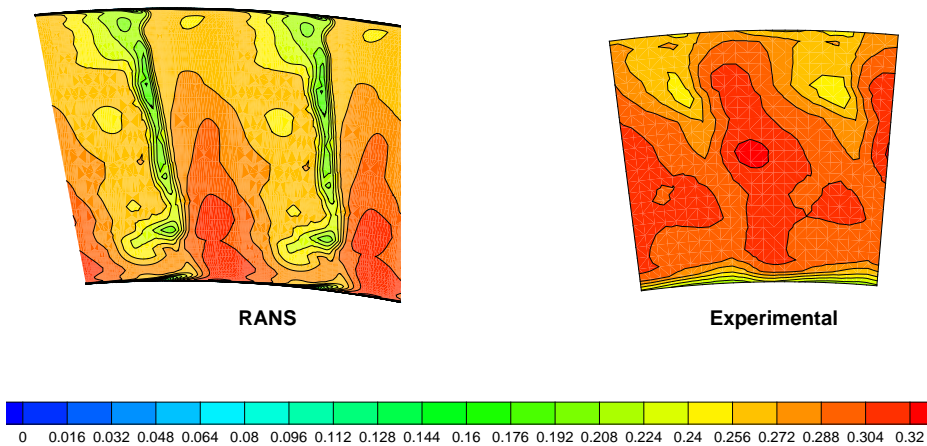


Figure 8.21: Cylindrical blade experimental and RANS Mach contours at station 4 and RPM=2000 for the three-stage turbine.

**Cylindrical Blade
PR=1.4 2600 RPM
Relative Mach Number at Station 3**

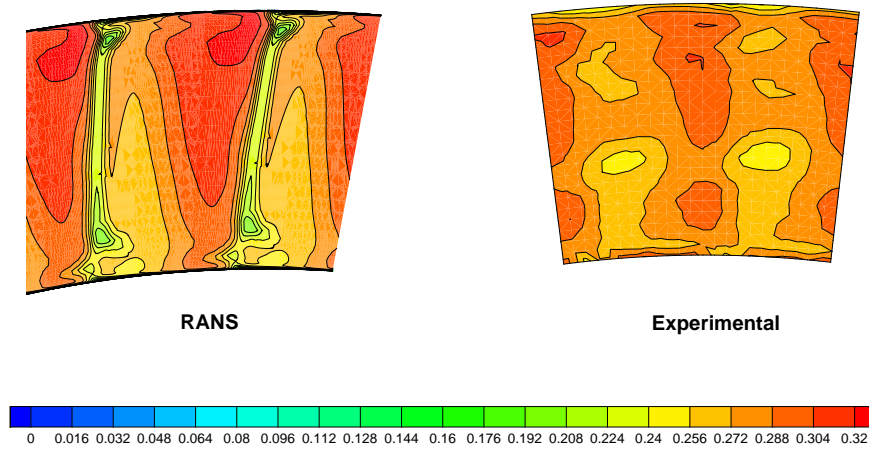


Figure 8.22: Cylindrical blade experimental and RANS relative Mach contours at station 3 and RPM=2600 for the three-stage turbine.

**Cylindrical Blade
PR=1.4 2000 RPM
Relative Mach Number at Station 3**

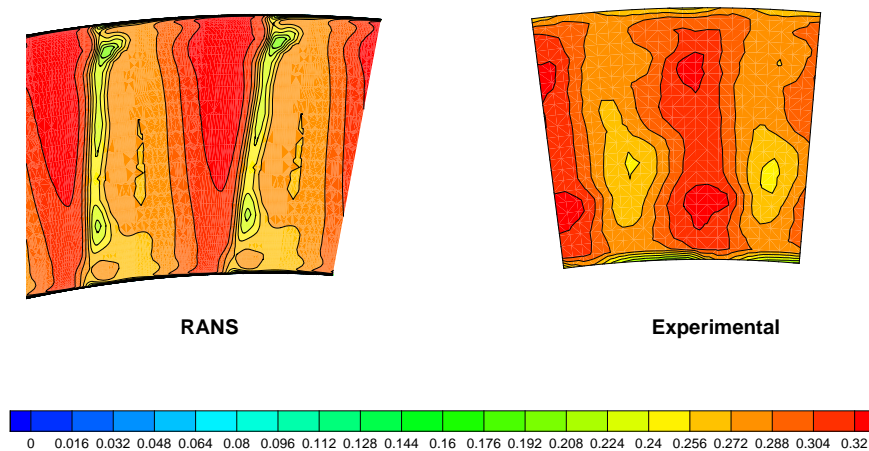


Figure 8.23: Cylindrical blade experimental and RANS relative Mach contours at station 3 and RPM=2000 for the three-stage turbine.

The discrepancy is consistent in figures showing total pressure (8.8 and 8.9) and relative

total pressure (8.10 and 8.11). Both absolute Mach number and relative Mach number for both 2000 RPM and 2600 RPM can be seen in figures 8.12 to 8.13. The mach numbers show a more favorable match with the experimental data, with near-wall high loss region showing the expected slight peaking of both absolute and relative mach numbers. These regions are localized to approximately 0-0.1 normalized span (near-hub) and 0.9-1.0 normalized span (near-tip).

A much better understanding of the differences between the high speed and low speed case can be made via comparing 2D contour plots of total pressure and relative total pressure at the rotor and stator exit points respectively, or station 3 and station 4 as they are termed here. Looking at the stator exit, station 4 figure 8.16 a contour plot of normalized total pressure for this location, with the RANS based results and the experimental results side by side. The normalization was done using the maximum and minimum values of total pressure in that plane, and are used to facilitate the comparison between the two cases. The normalization for total pressure can be defined using:

$$P_n = \frac{P - P_{min}}{P_{max} - P_{min}} \quad (8.1)$$

In the low speed case, both RANS and experimental results show slightly larger wake regions for the low speed (2000 RPM) case when compared to the higher speed 2600 RPM case. This is also observed when looking at a difference in station 4 Mach number, which is shown in figure 8.20 for the high speed 2600 RPM case and in figure 8.21 for the low speed 2000 RPM case. The wake shapes seem to be well predicted by the RANS solver when compared to the experimental contours, however it can generally be said that the CFD predicted wake regions are much smaller in size when compared to the data obtained using five-hole probes. This will generally lead to RANS over-predicting the efficiencies for both the cylindrical and 3D bowed blade series when compared to total-to-static efficiencies measured.

In the case of the rotor exit, station 3, figures 8.18 and 8.19 show 2D contour mappings

of the normalized relative total pressure, for the high speed and low speed cases respectively. In this case the normalization of the relative total pressure is done in a similar way as in equation (8.1):

$$P_{rn} = \frac{P_r - P_{rmin}}{P_{rmax} - P_{rmin}} \quad (8.2)$$

The wake structure for both the high speed case and low speed is very similar, which suggests that the loading of the first rotor blade row is similar at both high and low speed cases, this is a result of similar incidence for both cases. This pattern is also observed when comparing the relative Mach numbers at high (figure 8.22) and low shaft speed (figure 8.23). In the next section the 3D bowed 9600 series blade station results are examined.

8.3.2 3D Bowed Blade RANS Results

For the 3D bowed blade series simulations, the low speed case was performed at 1800 RPM while the higher speed case was run at 2800 RPM, figures 8.24 and 8.24 show the circumferentially averaged static pressures for stations 3,4 and 5, again one observes the discrepancy in pressure prediction and measured pressure at all three stations shown. This discrepancy is again at maximum, about 2 kPa. This discrepancy in pressure is observed in total pressure plots (figures 8.26 and 8.27) and relative total pressure (figures 8.28 and 8.29), where the RANS solver predicts averaged Mach numbers within about 0.05.

In the case of velocity both absolute and relative Mach numbers are shown for each of the high speed and low speed cases (figures 8.30 to 8.33). One thing of note, is that it becomes slightly clear that the 3D blade seems to create a smaller a secondary flow region or loss source when compared to the cylindrical blade at the high speed case, the effects of bowing will be discussed later in this chapter.

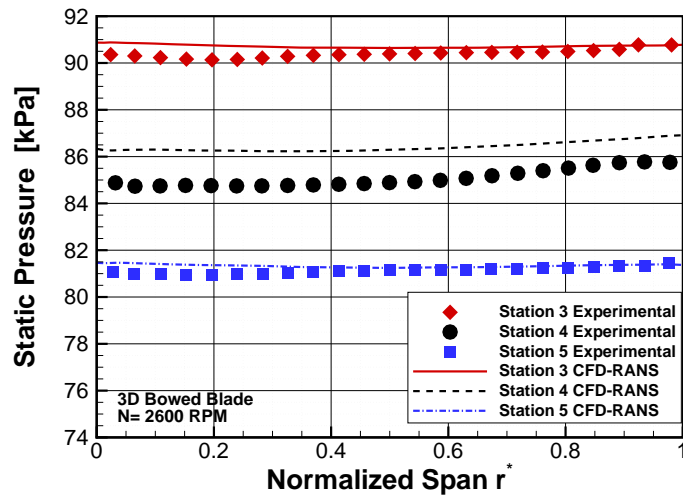


Figure 8.24: 3D bowed blade circumferentially averaged Static Pressure at 2600 RPM for Stations 3, 4 and 5 for three-stage turbine.

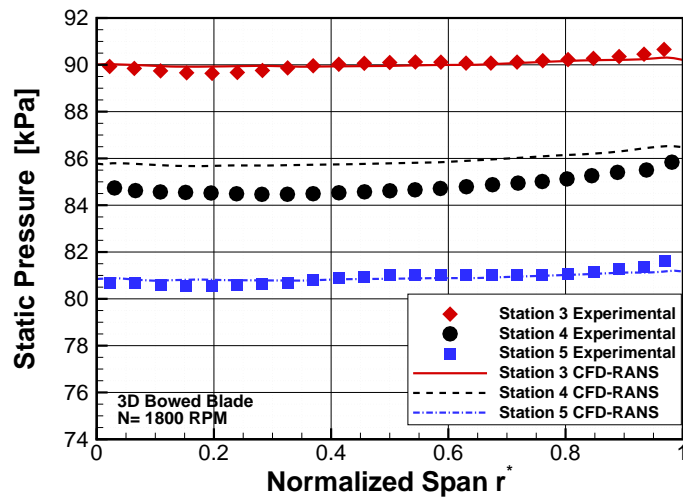


Figure 8.25: 3D bowed blade circumferentially averaged static pressure at 1800 RPM for stations 3, 4 and 5 for the three-stage turbine.

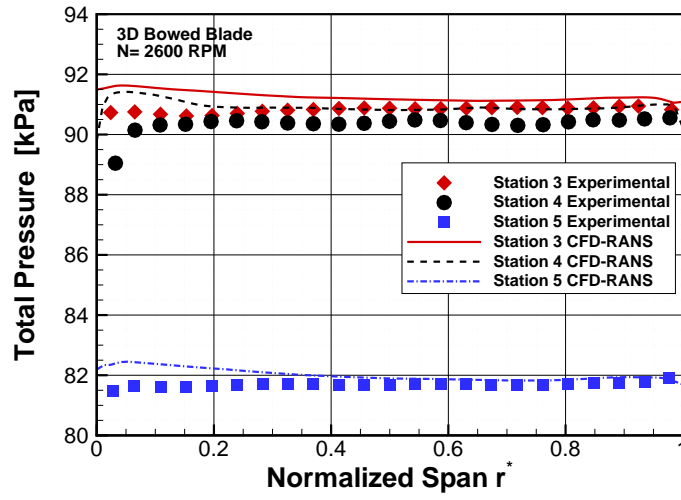


Figure 8.26: 3D bowed blade circumferentially averaged Total Pressure at 2600 RPM for Stations 3, 4 and 5 for the three-stage turbine.

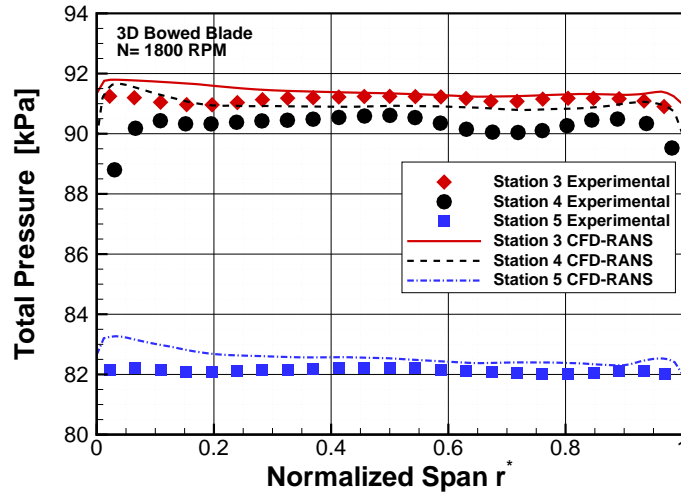


Figure 8.27: 3D bowed blade circumferentially averaged Total Pressure at 1800 RPM for Stations 3, 4 and 5 for the three-stage turbine.

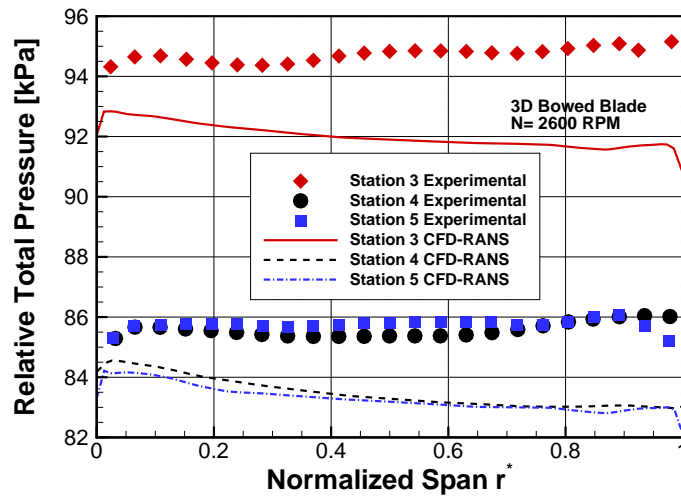


Figure 8.28: 3D bowed blade circumferentially averaged Relative Total Pressure at 2600 RPM for Stations 3, 4 and 5 for the three-stage turbine.

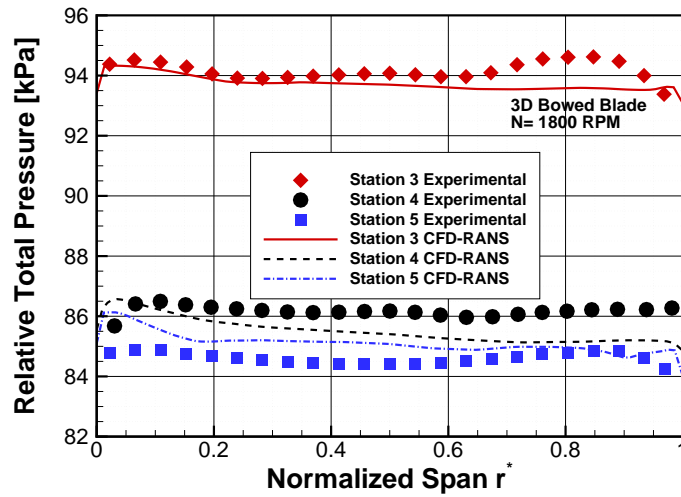


Figure 8.29: 3D bowed blade circumferentially averaged Relative Total Pressure at 1800 RPM for Stations 3, 4 and 5 for the three-stage turbine.

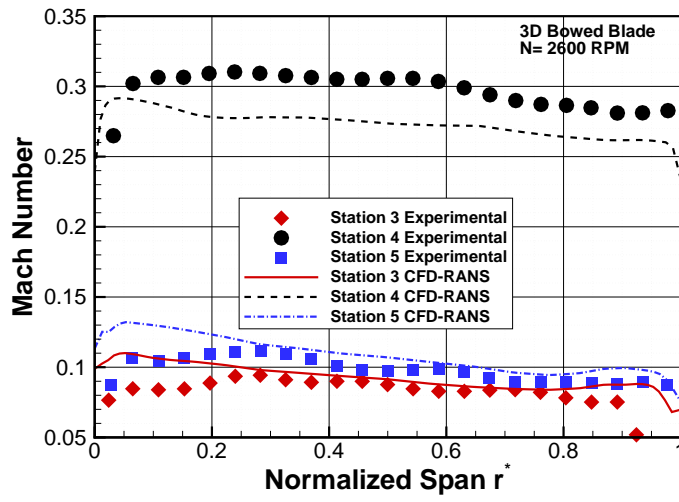


Figure 8.30: 3D bowed blade circumferentially averaged Absolute Mach Number at 2600 RPM for Stations 3, 4 and 5 for the three-stage turbine.

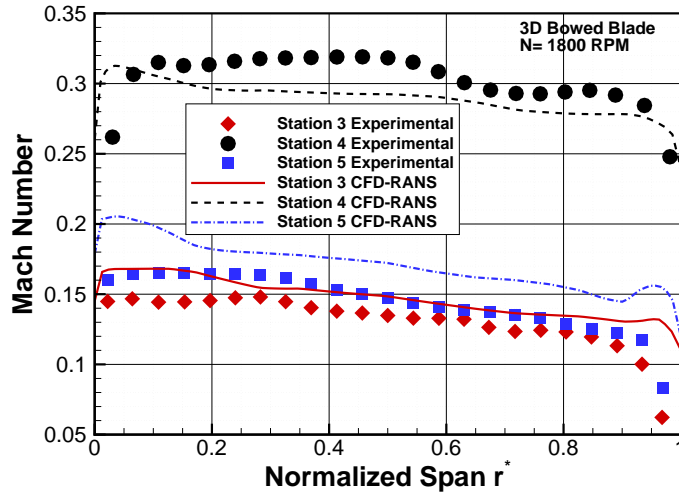


Figure 8.31: 3D bowed blade circumferentially averaged Absolute Mach Number at 1800 RPM for Stations 3, 4 and 5 for the three-stage turbine.

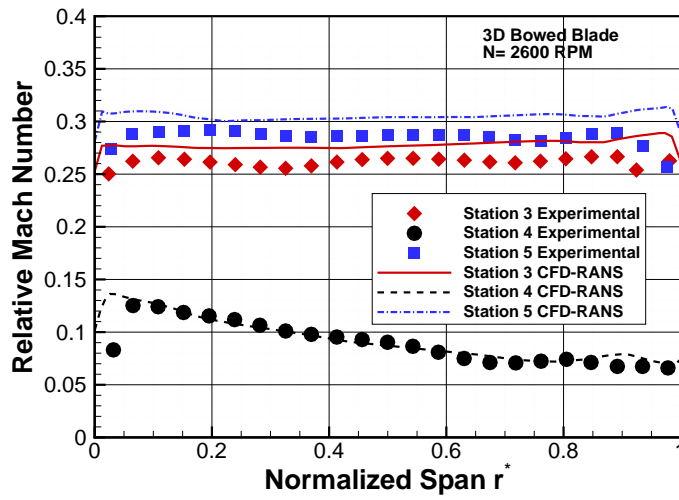


Figure 8.32: 3D bowed blade circumferentially averaged relative Mach number at 2600 RPM for stations 3, 4 and 5 for the three-stage turbine.

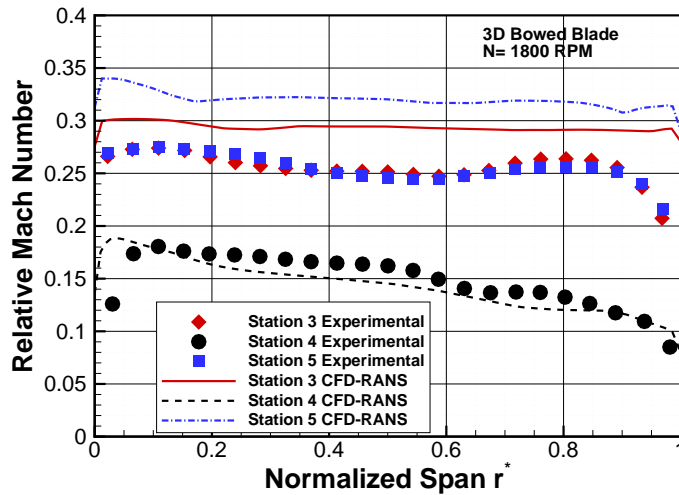


Figure 8.33: 3D bowed blade circumferentially averaged relative absolute Mach number at 1800 RPM for stations 3, 4 and 5 for the three-stage turbine.

3D Bowed Blade
PR=1.4 2600 RPM
Normalized Total Pressure at Station 4

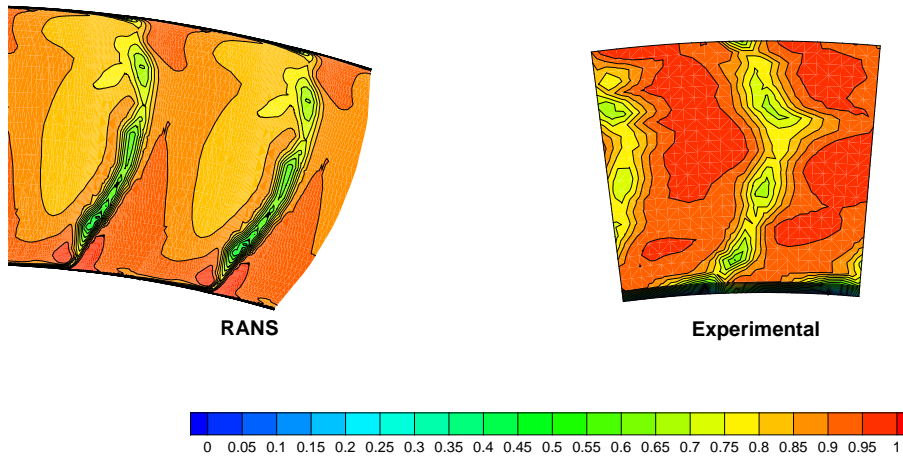


Figure 8.34: 3D bowed blade experimental and RANS normalized Total Pressure Contour at Station 4 and RPM=2600 for the three-stage turbine.

3D Bowed Blade
PR=1.4 1800 RPM
Normalized Total Pressure at Station 4

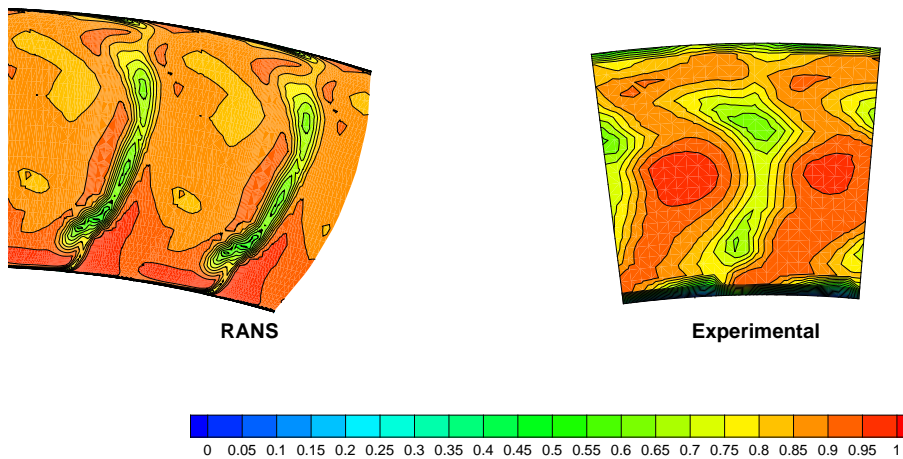


Figure 8.35: 3D bowed blade experimental and RANS normalized total pressure at station 4 and RPM=1800 for the three-stage turbine.

3D Bowed Blade
PR=1.4 2600 RPM
Normalized Relative Total Pressure at Station 3

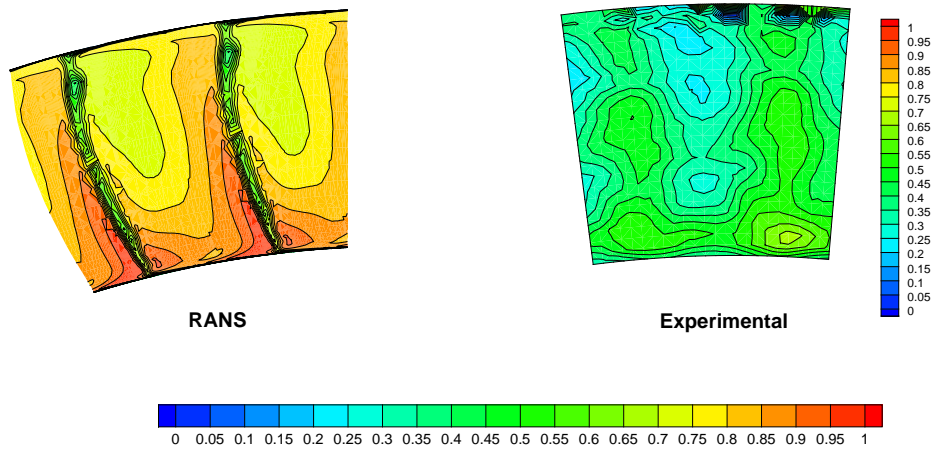


Figure 8.36: 3D bowed blade experimental and RANS normalized relative total pressure contours at station 3 and RPM=2600 for the three-stage turbine.

3D Bowed Blade
PR=1.4 1800 RPM
Normalized Relative Total Pressure at Station 3

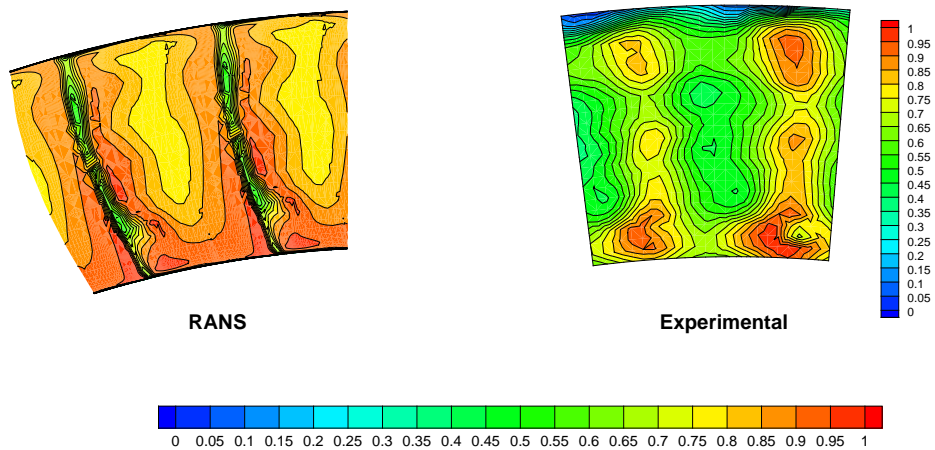


Figure 8.37: 3D bowed blade experimental and RANS normalized relative total pressure contour at station 3 and RPM=1800 for the three-stage turbine.

3D Bowed Blade
PR=1.4 2600 RPM
Mach Number at Station 4

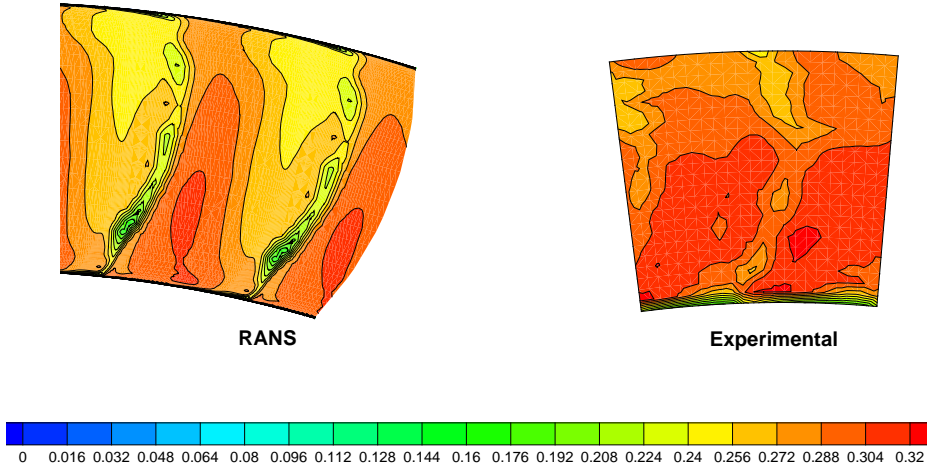


Figure 8.38: 3D bowed blade experimental and RANS Mach contours at station 4 and RPM=2600 for the three-stage turbine.

3D Bowed Blade
PR=1.4 1800 RPM
Mach Number at Station 4

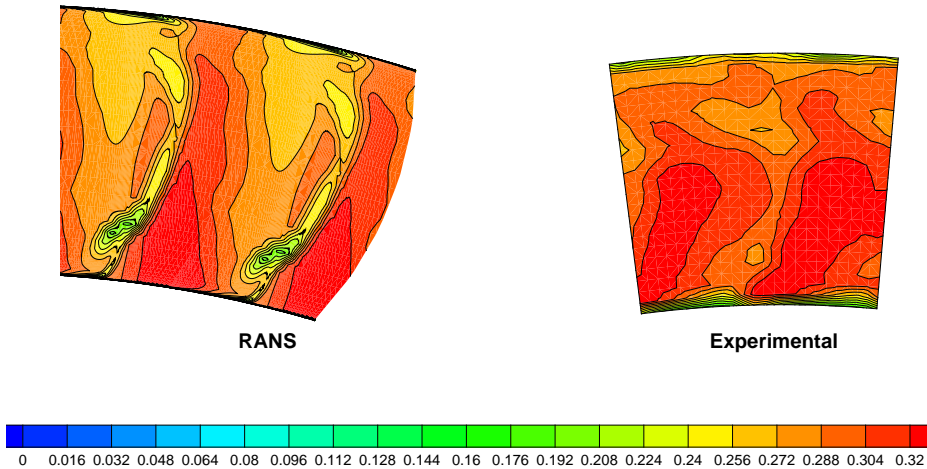


Figure 8.39: 3D bowed blade experimental and RANS Mach contours at station 4 and RPM=1800 for the three-stage turbine.

3D Bowed Blade
PR=1.4 2600 RPM
Relative Mach Number at Station 3

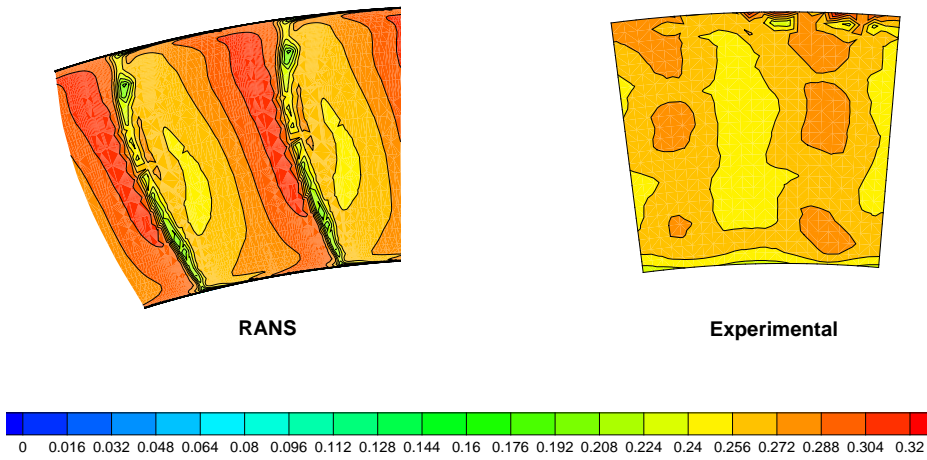


Figure 8.40: 3D bowed blade experimental and RANS relative Mach contours at station 3 and RPM=2600 for the three-stage turbine.

3D Bowed Blade
PR=1.4 1800 RPM
Relative Mach Number at Station 3

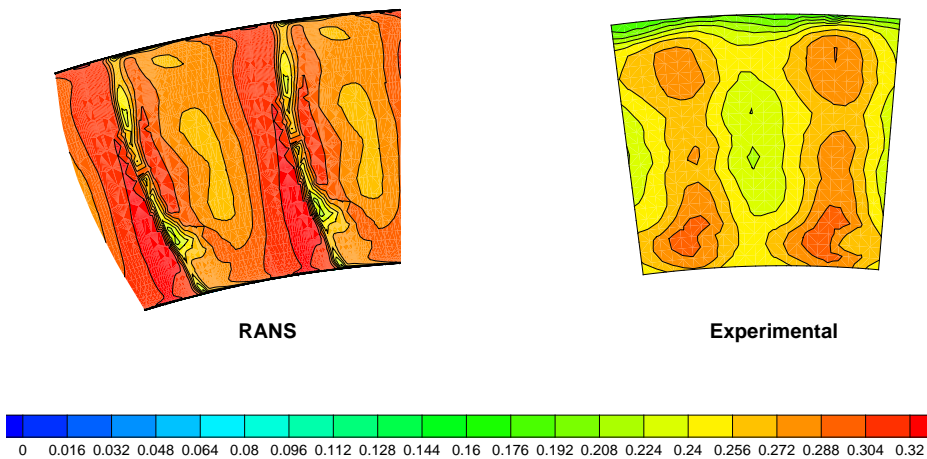


Figure 8.41: 3D bowed blade experimental and RANS relative Mach contours at station 3 and RPM=1800 for the three-stage turbine.

Normalizing the total and relative total pressure using equations (8.1) and (8.2) 2D contour maps of total pressure can be compared for both the high speed and low speed cases. Figures 8.34 and 8.35 show the normalized total pressure contour at station 4 for both the high speed and low speed case. The bowed nature of the stator blade is observed in the new wake shape observed in both the RANS and experimental wake regions. As shown a larger wake regions with higher loss are observed at the lower speed case which is due to increased incidence on the second stator blade row. Similar to what is observed earlier in the 2-stage analyses, and the cylindrical blade analyses, the CFD based wake is much smaller in size when compared to the experimentally determined wake. In the case of the relative pressure at station 3, shown in figure 8.36 for the high speed case, and 8.37 for the low speed, case the wakes at the exit of the first rotor blade show little differences between the high speed case and the lower speed case, this is most likely due larger changes in loading for the second stage at the lower speed case when compared to the higher speed case, whereas it appears that the loading in the first stage blade rows is not affected by the change in shaft speed.

When looking at the Mach number profiles at the station 3 and 4 locations in figures 8.38 to 8.41, the same trends are observed, namely, smaller wake regions predicted by CFD, the small impact of shaft speed on the wake shape in station 3, whereas for station 4 the wake region size increases at the lower speed point with larger loss regions in the casing region. Overall, the CFD RANS solver shows good correlation in 2D loss trends at both design points, and would give the designer confidence in its use as a tool to compare the impact of changing the shaft speed on the overall machine performance as well as the specific wake deficit regions.

8.4 Performance Results and 3D Blade Effects

The use of compound lean in blade design, or specifically positive lean as is the case with the 9600 series blades, is found to be advantageous in terms of achieving higher efficiency. Positive lean creates higher pressure in the end-wall regions when compared to the mid-span pressure. This higher pressure zone creates a general movement of flow

towards the mid-span which aids in reducing the intensity of the end-wall vortices. In a series of cascade tests, Matsunuma et al. [98] found that the overall losses of the 2D cascade is similar to the losses in the compound lean cascade, but the higher loss regions are moved closer to the mid-span region than being localized to the end-walls. Most blade profiles are designed at the meanline, which provides a optimized profile for the meanline, however at the hub and tip locations, if this profile is extruded to generate a cylindrical or 2D blade, a large incidence will be imparted at the blade based on radial equilibrium. This incidence brings about the near wall vortices that cause the secondary flow based losses that are highly pronounced in low aspect turbine blades. The effects of blade lean and sweep are discussed in detailed by Denton and Xu [99].

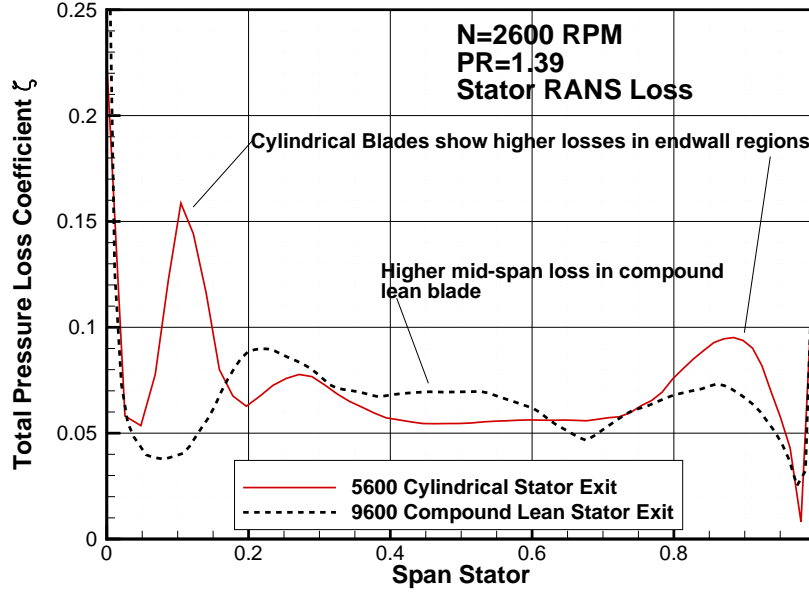


Figure 8.42: Total pressure loss coefficient station4 at 2600 RPM for both cylindrical and 3D bowed blades for the three-stage turbine.

As shown in figure 8.42 the impact of using compound lean is illustrated, whereby, the high loss regions found in the cylindrical blades (in red) are reduced in the 3D blades, whilst slightly increasing the mid-span loss. Figure 8.43 allows one to visualize the loss

regions in a clearer fashion. The entropy regions for the cylindrical blades at the low speed point of 1800 rpm show larger loss zones in the endwall regions, whilst the the compound blades observe a higher mid-span region loss.

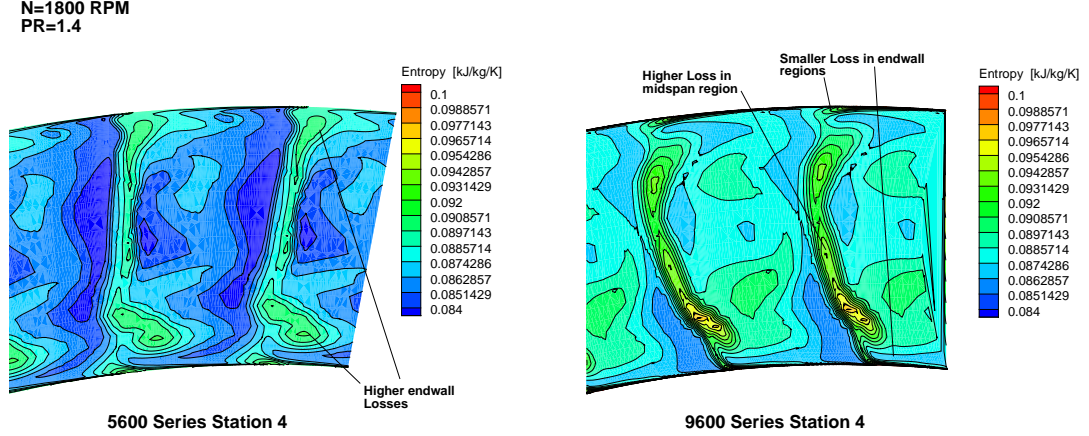


Figure 8.43: Station 4 entropy contours at 1800 RPM for both cylindrical and 3D bowed blades for the three-stage turbine.

The overall effect on performance for both blade designs is shown in figures 8.4 and 8.4 which show the total-to-static efficiency at varying shaft speeds for both RANS and experimental data. As expected, the RANS solver over-predicts the efficiency for both cylindrical and 3D bowed blades by a significant amount, to just about 4 points at the low speed cases, and 1 point at the high speed case. This is expected, as the RANS solver has shown consistently that the use of the mixing plane method, and the SST model significantly shows smaller wake regions at the exit station of each blade row examined. The RANS solver is more deficient in its prediction at the lower speed points than at the higher speed points. This is because the lower speed points introduce greater incidence on the blade rows, which will result in increased secondary flow based losses as well as higher profile losses due to the increased incidence. CFD struggles to accurately predict the increased extent of the secondary flow losses at the lower speed points.

Moreover, the use of positive lean results in significantly higher efficiency across each

testing point, experimental results showing on average an increase of 0.75 points of efficiency when using the compound lean blade design, whilst the CFD solver showing a more optimistic 1.22 average improvement in efficiency points. Overall the RANS solver results show relatively good matching particularly for the cylindrical and 3D bowed series blades, when compared to the Doosan based design discussed earlier. However the concern remains that the performance over-prediction is significant from an absolute point view, however it seems that from a relative, or design point to design point approach the CFD provides valuable information on how the particular design will perform under different operating conditions. Another approach worth looking at, which requires significantly larger computing resources is the URANS or unsteady based modeling, the results of this approach are discussed in the next section.

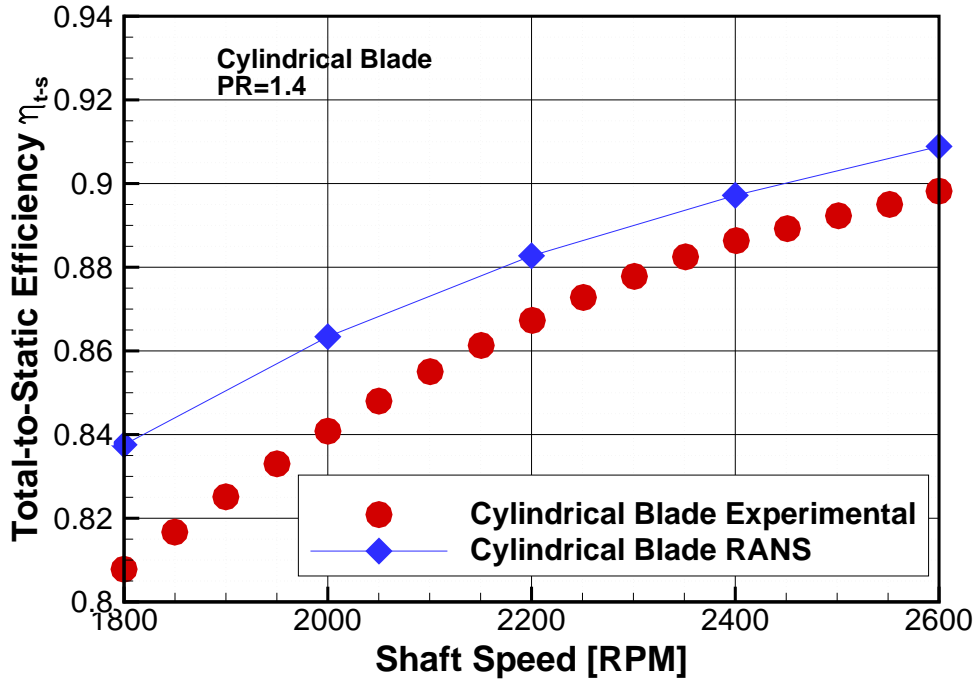


Figure 8.44: Experimental and RANS based total-to-static efficiency η_{t-s} at varying shaft Speeds for cylindrical blades. Pressure ratio=1.4 for the three-stage turbine.

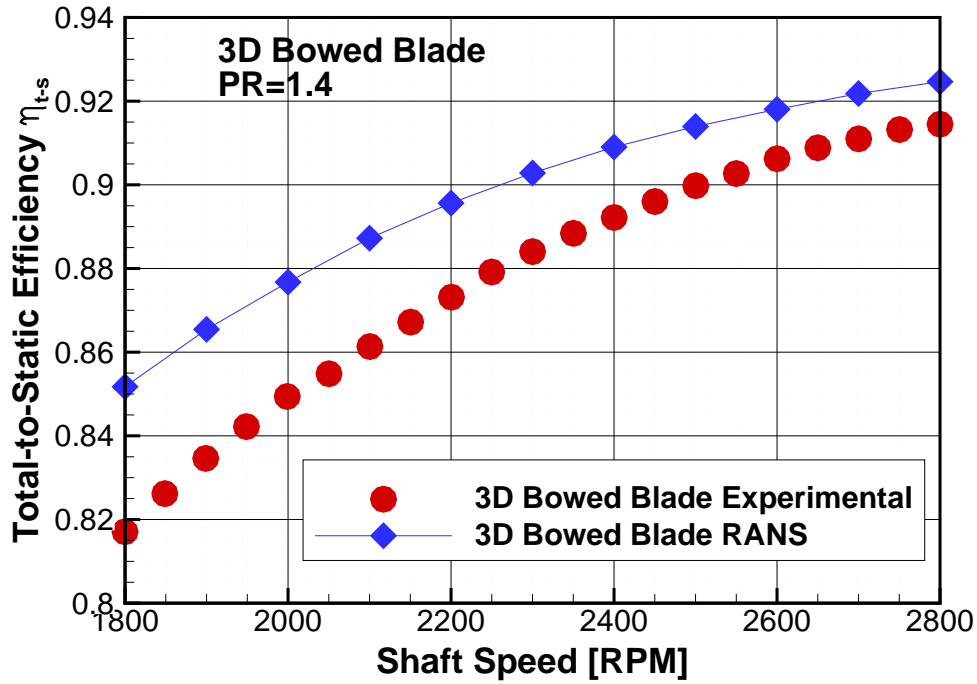


Figure 8.45: Experimental and RANS based total-to-static efficiency η_{t-s} at varying shaft speeds for 3D bowed blades. Pressure ratio=1.4 for the three-stage turbine.

8.5 URANS Based Results

In chapter 7 an unsteady based solver was used to simulate a 2-stage high pressure turbine design and compared to both experimental and a RANS based solver. The use of URANS provides further insight into the complex flow exhibited in turbomachinery, URANS will allow the analyst to observe the various rotor/stator wake interactions, as well as avoid the use of mixing planes, which create a mixing based entropy increase at each interface which potentially provides an unrealistic model of the physics between rotating and stationary reference frames. However, the use of unsteady based CFD or URANS techniques presents new challenges which include:

- Area matching between rotating and stationary domains

- Increased storage requirements to save each time step results
- Increased computational requirements and computational times

Area matching, refers to the need to have equal area patches between the stator exit and the downstream rotor inlet. In the case of the RANS solver, the mixing plane does not require an equal area patch across the interface between stator and rotor. For the mixing plane technique, each blade row a steady state solution is calculated, and the two adjacent rows are coupled via an exchange of flow field variables. Flow field data is averaged circumferentially for both frames of references and then passed to the adjacent blade row as a boundary condition. Flow variations in the pitchwise direction can be neglected at the interface, and thus the assumption is that the relative flow field in successive blade rows is steady. As a result of averaging process at the interface, any time dependent phenomena, or any stator/rotor interaction is neglected as it will be averaged out at the interfaces. Moreover, the mixing plane model introduces an error generated by the artificial mixing process that is applied at the interface plane. On the other hand in the case of unsteady analyses, the solution from an upstream blade is directly mapped onto the subsequent downstream blade row 'inflow' patch. Mapping can occur via a direct 1:1 node matched grid, or in some instances various grid-to-grid interpolation techniques are used for mesh interfaces that are not matched in terms of discretization. However, the unsteady interface requires that the annular area across the patch is equal.

Due to computational resource limits, as well as storage limits, and that the blade numbering shown in table 8.1 will not provide a common multiple number across the three blade stages (6 blade rows), it was decided to fix the blade number for all blades to 50 blade instances. This allows the use of a one blade instance mesh for each blade row. The two mesh domains in figure 8.3 for the cylindrical blade case and figure 8.4 for the 3D bowed blade was modified and used for the URANS analysis. Both a low speed and high speed case was run for the compound lean blade and the cylindrical blade. The boundary conditions for each case are listed in table 8.2 and are the same boundary conditions ex-

tracted from experimental data and used by the RANS cases, the turbulence intensity for all CFD models was set to 2% and the RANS cases were used as an initial step for all the URANS simulations.

The focus of the URANS based cases was to model the low speed cases which showed the largest discrepancy between RANS and experimental efficiency. The aim was to see if the unsteady based URANS would provide a better prediction of machine efficiency at the off-design case. In unsteady RANS mode, the commercial solver used for these cases reverts to a second order backward Euler time based scheme, along with the second-order upwinding discretization scheme. The boundary conditions used for the low speed 1800 rpm case and the high speed 2600 rpm (both at a pressure ratio of 1.4) is listed in table 8.3, the time step was set to 1.04167×10^{-6} seconds. With every other time step saved, this provides about 300 steps per blade passage period. The interfaces between blade rows was set to a mapped grid-to-grid unsteady interface that does not use any kind of circumferential averaging. Convergence history for the RANS cylindrical blade case and the URANS 3D bowed blade case is shown in figures 8.46 and 8.47 respectively. As shown the RANS and URANS solvers show good convergence characteristics for both models, it was noted that with higher refinement a much larger number of iterations was required to reach the 1×10^{-6} convergence criteria for all equations solved, moreover the convergence history showed a oscillatory trend which suggests the presence of time-dependent flow characteristics where a URANS model is better suited. The URANS solver was set to use 10 inner iterations, to achieve the required convergence criteria after the rotating domains have been clocked as part of the time-marching process.

Table 8.3: Unsteady boundary conditions for 1800 RPM and 2600 RPM cases.

Unsteady Boundary Conditions	1800 rpm	2600 rpm
Reference Pressure [kPa]	101.261	100.975
Total Pressure Inlet [kPa]	100.496	100.271
Static Pressure Outlet [kPa]	71.654	72.021
Total Temperature Inlet [K]	323.48	324.0
Total Temperature Outlet [K]	298.3	297.05
Inlet Turbulence Intensity [%]	2.0	2.0
Total Mesh Size [nodes]	4.8×10^6	4.8×10^6
Time Step [sec]	1.04167×10^{-6}	1.04167×10^{-6}

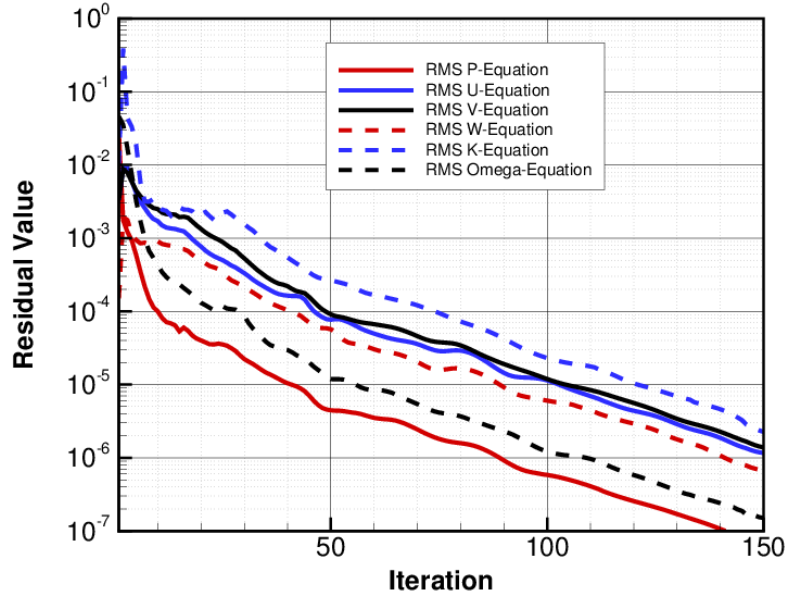


Figure 8.46: RANS root mean square residual errors versus iteration number for 1800 rpm cylindrical blade case, using the SST shear stress transport turbulence model.

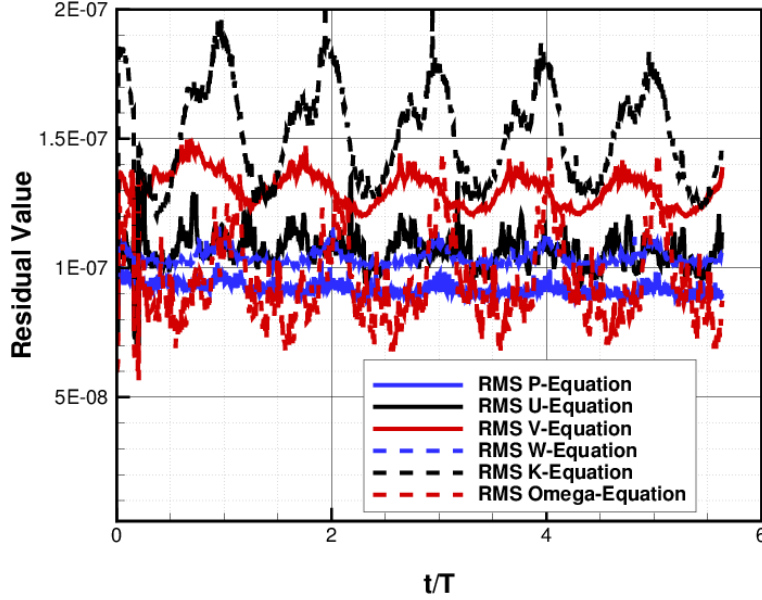


Figure 8.47: URANS root mean square residual error versus blade passage period t/T for 1800 rpm cylindrical blade case with the SST shear stress transport turbulence model.

8.5.1 Cylindrical Blade URANS Results

At this point, a comparison between the URANS and RANS predictions is made. Although the unsteady aspect of the model will allow one to model all time-dependent flow characteristics such as rotor wake impingement, it must be stated that in order to accommodate the computational needs, the URANS grid was coarsened significantly which could impact the quality of the prediction negatively.

Figures 8.48 to 8.50 show the circumferentially and temporally averaged total pressure, relative total pressure and static pressure at stations 3,4 and 5 at 1800 RPM for the cylindrical blade, compared to the RANS results. It is clear that both RANS and URANS predict similar pressure trends from the hub to tip regions, with the averaged transient data showing more of a smoothed profile. The averaging technique here was the consistent averaging technique as explained in by Schobeiri in [8]. However, as aforementioned the smoother profiles of the URANS data is most likely due to the coarsened mesh that was

used for the URANS cases as a result of the high computational and storage requirements of the transient models. This pattern is also exhibited in the absolute Mach and relative Mach number profiles shown in figures 8.51 and 8.52, where in the near wall regions the URANS profiles show less impact on velocity due to end wall secondary flow effects, the matching between both RANS and URANS shows good consistency as well in the stator 2 and rotor 2 exit total pressure loss profiles shown in 8.53. A clearer comparison between both techniques can be shown by comparing temporally averaged contours with the their corresponding RANS counterparts, this was done for both station 3 and 4, and by showing total pressure in the rotating and stationary references as well as the entropy at the exit of those blade rows. Looking at entropy contours is another way to assess the loss incurred in the blade, and specifically the loss due to secondary effects. This was done in figure 8.54 for the total pressure maps and figure 8.55 for the entropy. In the total pressure contours as well as the entropy contours one can see a large similarity between both the URANS and rans cases, however, the RANS profiles show large loss spots in the near wall regions with a more defined shape versus the temporally averaged URANS profiles.

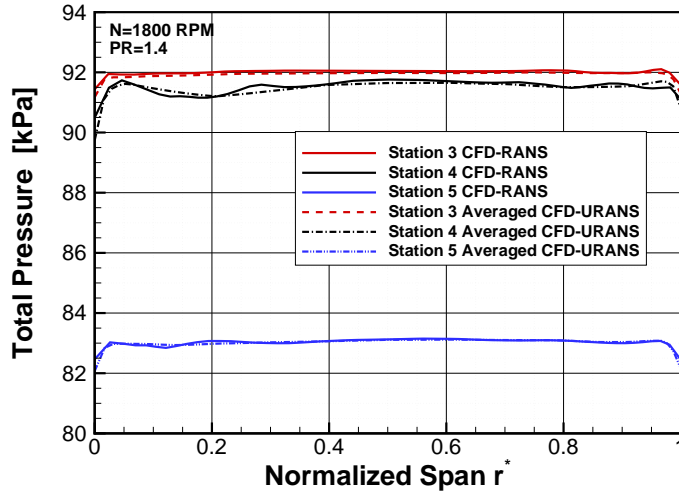


Figure 8.48: Temporally and circumferentially averaged total pressure at station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.

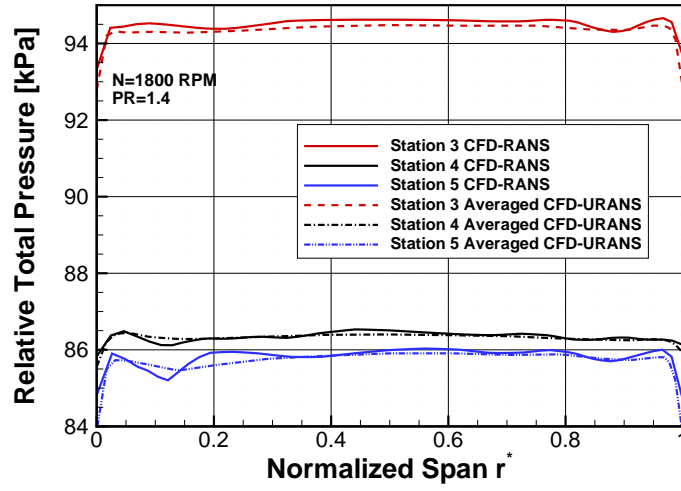


Figure 8.49: Temporally and circumferentially averaged relative total pressure at station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.

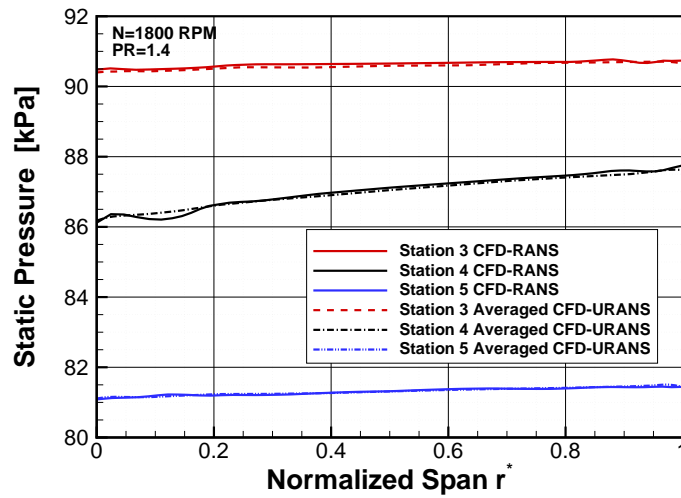


Figure 8.50: Temporally and circumferentially averaged static pressure at station 3,4, and 5 for the cylindrical blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.

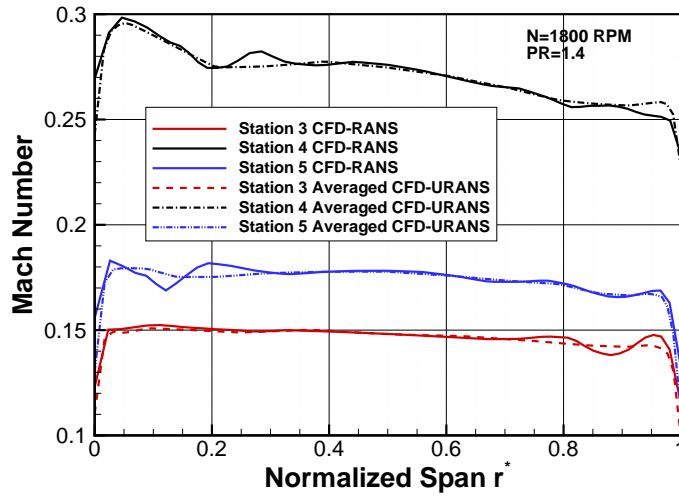


Figure 8.51: Temporally and circumferentially averaged Mach number at Station 3,4, and 5 for the cylindrical blade at 1800 RPM $PR=1.4$ compared to RANS result for the three-stage turbine.

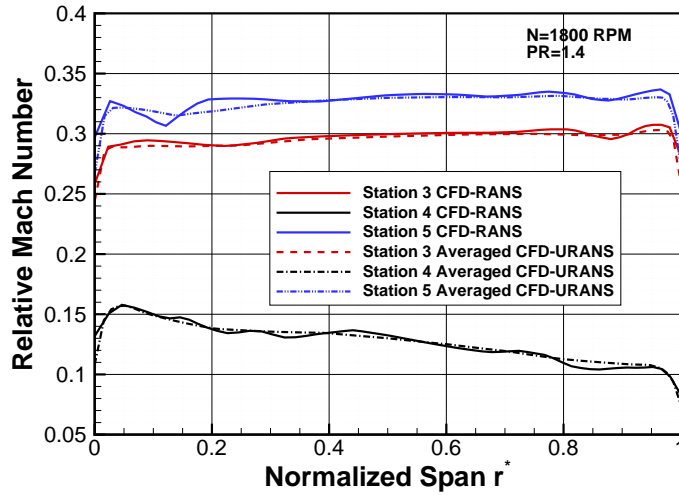


Figure 8.52: Temporally and circumferentially averaged relative Mach number at Station 3,4, and 5 for the cylindrical blade at 1800 RPM $PR=1.4$ compared to RANS result for the three-stage turbine.

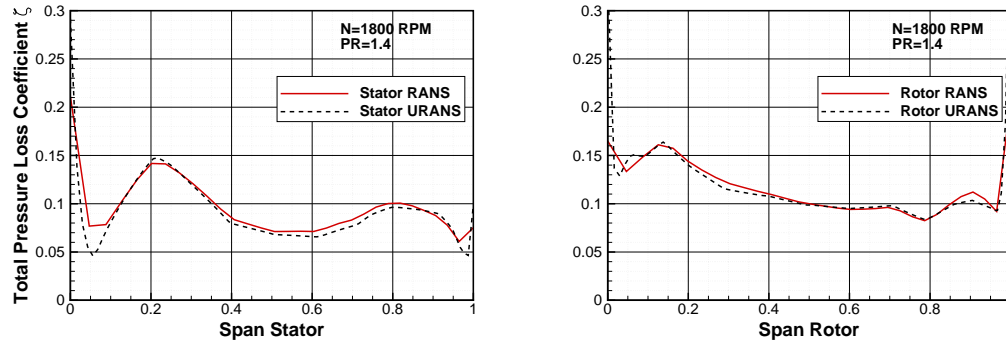


Figure 8.53: Temporally and circumferentially URANS and RANS based rotor and stator total pressure loss coefficient for the cylindrical blade at 1800 RPM PR=1.4 for the three-stage turbine.

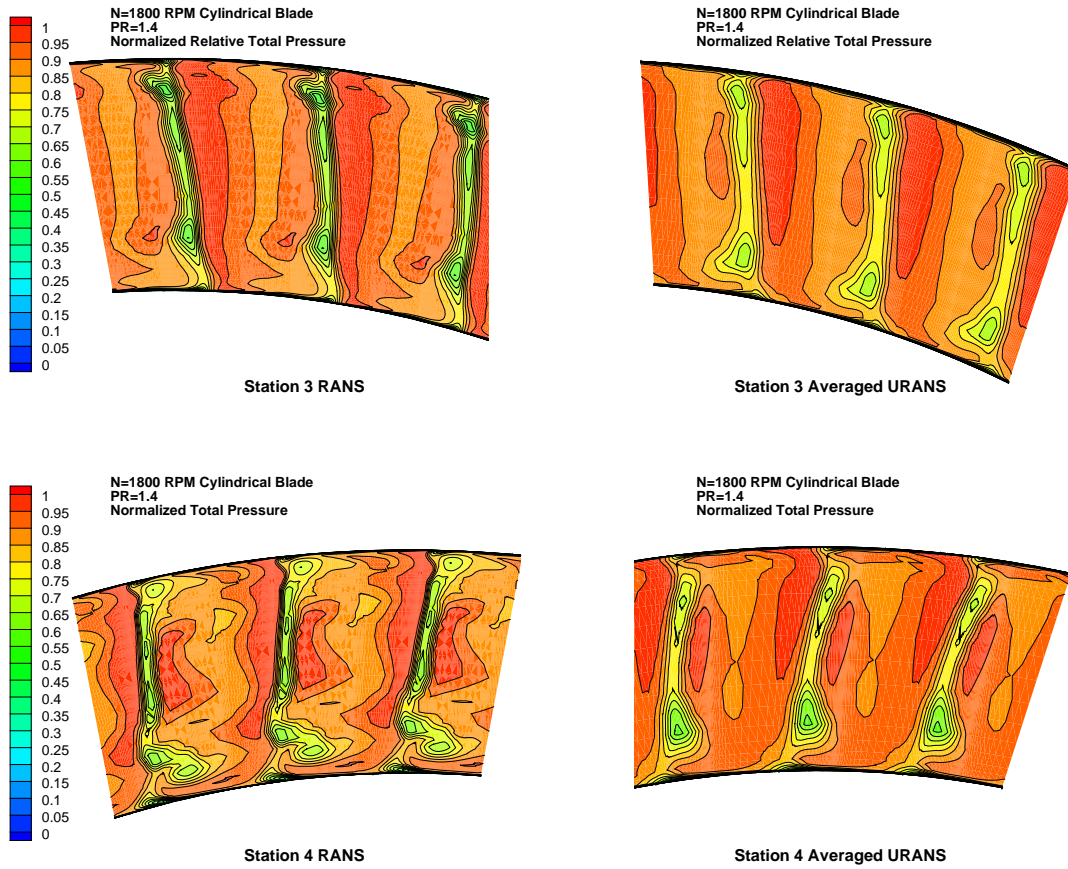


Figure 8.54: Temporally averaged URANS total pressure contours at stations 3 and 4 versus corresponding RANS data at N=1800 RPM and PR=1.4 for the three-stage turbine.

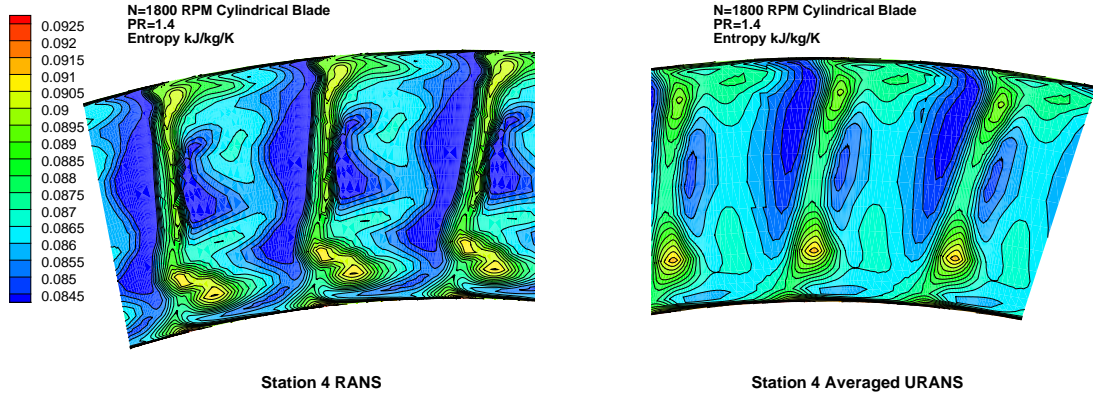


Figure 8.55: Temporally averaged URANS entropy [$kJ/kg K$] contours at stations 3 and 4 versus corresponding RANS data at $N=1800$ RPM and $PR=1.4$ for the three-stage turbine.

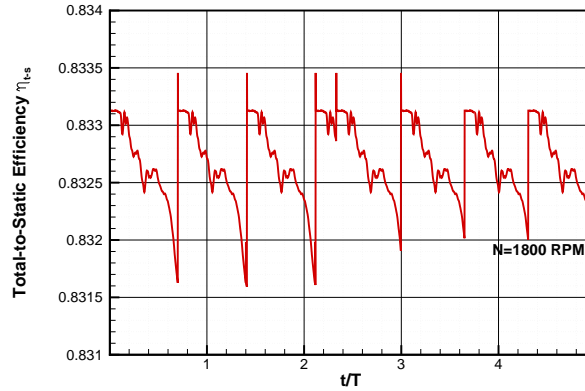


Figure 8.56: Cylindrical blade URANS total-to-static efficiency versus blade passage time for 1800 RPM.

Plotting the progression of massflow at the model exit in figure 8.57 and the total-to-static efficiency in figure 8.56 one can see the change in the values as a function of the blade wake passage period. As the rotor blade pass through one passage instance the massflow

increases, whilst the efficiency drops. A more in-depth look into the unsteady effects will be addressed later in this chapter.

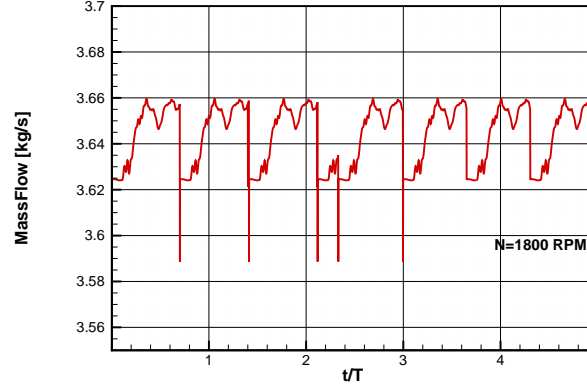


Figure 8.57: Cylindrical blade URANS massflow [kg/s] versus blade passage time for 1800 RPM.

8.5.2 3D Bowed Blade URANS Results

Similarly for the 3D bowed blade series compound lean rotor and stator blade design, an analysis of the URANS, RANS and in this case experimental data is shown. As with the cylindrical blade RANS and URANS data, and primarily because both simulations used the same boundary conditions, both the temporally averaged URANS data and the RANS data shows similar profiles for pressure and velocities and stations 3, 4 and 5 (figures 8.58 to 8.61). Subtle differences between the loss profiles for the 3D bowed blade and the cylindrical blade can be observed, as shown in figure 8.62 specifically for the stator exit location, in accordance with the purpose of blade positive lean, the losses in the mid span of the 3D bowed blade have increased, whereas the loss in the near-wall regions have decreased. In the case of the rotor exit, it appears that for both URANS and RANS, the 3D bowed blade near-wall losses have increased when compared to the cylindrical blades as shown in figure 8.53.

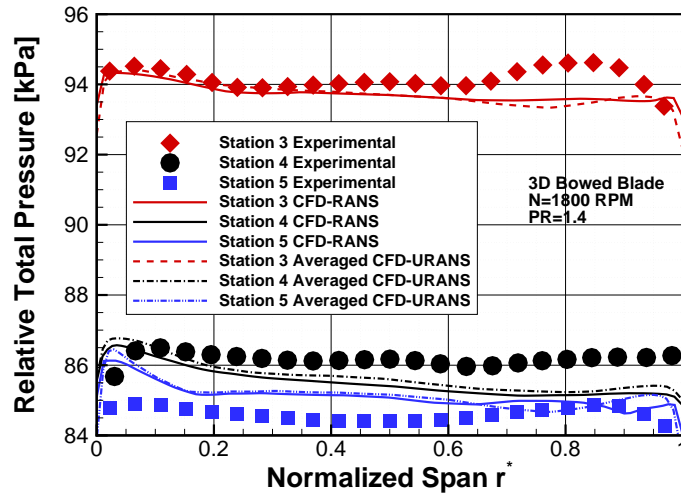


Figure 8.58: Temporally and circumferentially averaged relative total pressure at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.

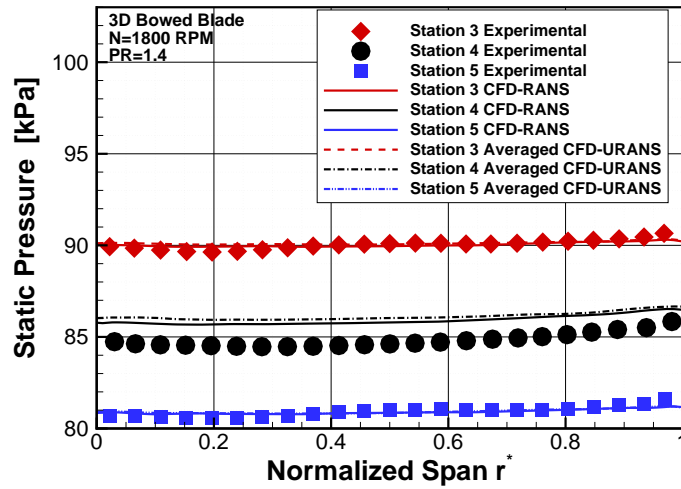


Figure 8.59: Temporally and circumferentially averaged static pressure at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.

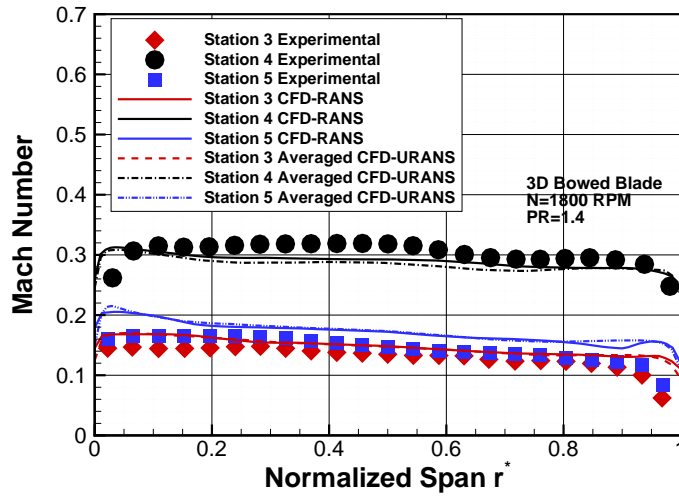


Figure 8.60: Temporally and circumferentially averaged Mach number at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.

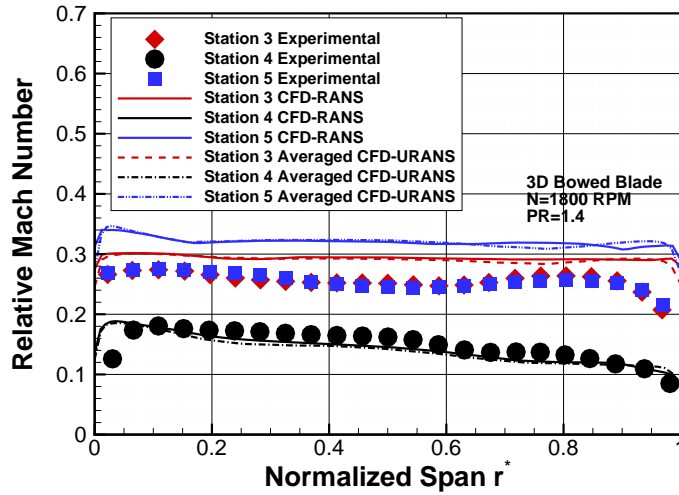


Figure 8.61: Temporally and circumferentially averaged relative Mach number at station 3,4, and 5 for the 3D bowed blade at 1800 RPM PR=1.4 compared to RANS result for the three-stage turbine.

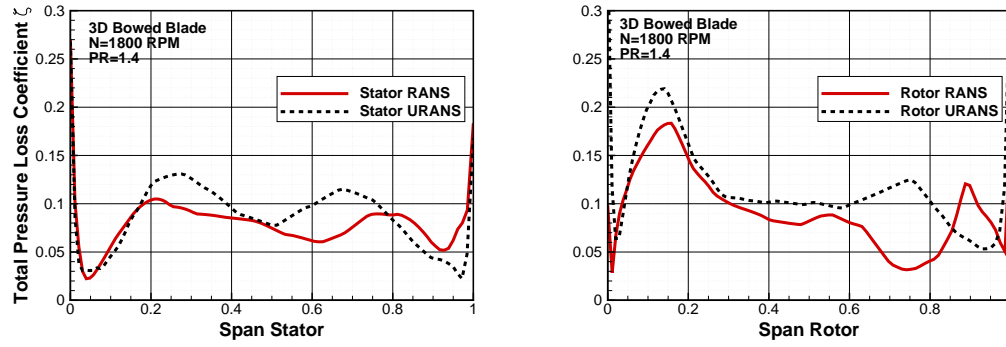


Figure 8.62: Temporally and circumferentially averaged URANS and RANS based rotor and stator total pressure loss coefficient for the 3D bowed blade at 1800 RPM PR=1.4 for the three-stage turbine.

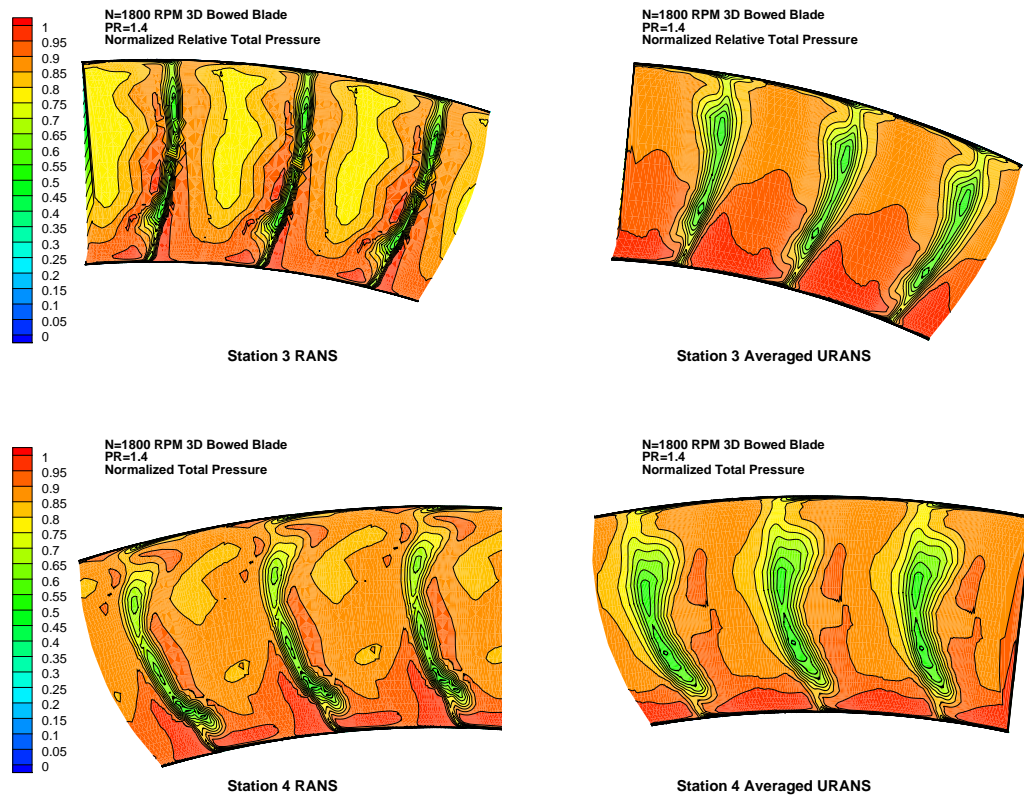


Figure 8.63: 3D Bowed Blade temporally averaged URANS total pressure contours at stations 3 and 4 versus corresponding RANS data at N=1800 RPM and PR=1.4 for the three-stage turbine.

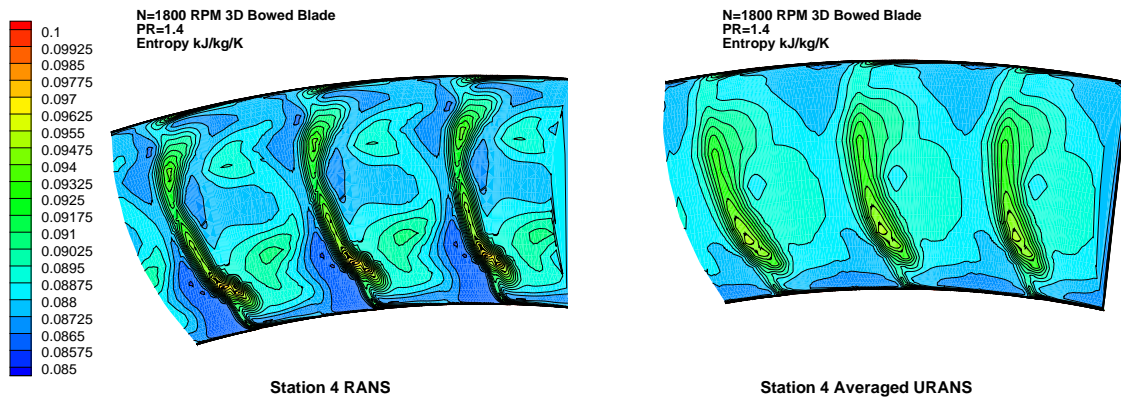


Figure 8.64: 3D bowed blade temporally averaged URANS entropy $[kJ/kg K]$ contours at stations 3 and 4 versus corresponding RANS data at $N=1800$ RPM and $PR=1.4$ for the three-stage turbine.

Considering the contour plots for total pressure at station 3 and 4 (figure 8.63), the aim of the compound lean is well illustrated for both URANS and RANS profiles. Comparing to figure 8.54 one sees a larger wake deficit in the mid span region that does not extend clearly to the near-wall regions as it does in the cylindrical blade profiles. The wake is thicker however dissipates in the near wall regions, where a significant degradation of the blade performance occurs. This again is illustrated clearly in the static entropy contour shown in figure 8.64.

8.5.3 3D Unsteady Effects

As stated previously, in order to run a RANS model an interface model is required to pass exit information from one blade row to subsequent downstream blade. Such interface models like the ones implemented by Denton and Xu [99] will average out any time dependent wake effects that would be impinging on the subsequent blade row.

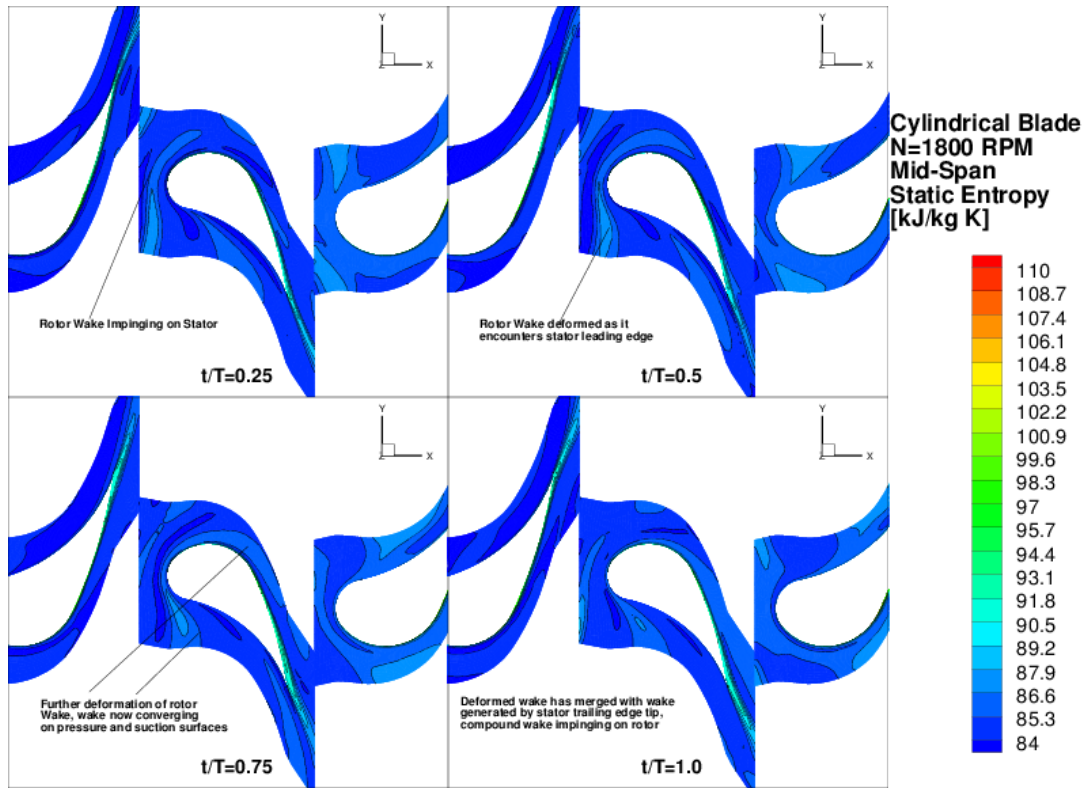


Figure 8.65: Cylindrical blade 1800 RPM mid-span entropy contours at various blade passage instants (rotation of rotor blades not shown).

A transient or URANS simulation using full mapped grid interfaces between the blade rows will not average out any time dependent wake interaction and provides further insight into how the wake interaction physics influence blade losses. The URANS cases run for cylindrical and 3D bowed blade series, were both run at the low speed 1800 rpm or off-design cases. Figures, 8.65 and 8.66, show various time snapshots of mid-span entropy as the rotor blades pass through one blade passage period. The trailing edge wakes from upstream blade row (second stage rotor) impinge onto stator leading edge. As the rotation proceeds, the wake is shown to be stretched and begins wrapping around the leading edge, the wake gravitates towards both the pressure surface and the suction surfaces warping in shape and eventually converging onto the stator blade's own trailing edge wake to create

a 'compound wake', which includes contributions from the stator trailing edge as well as the deformed upstream wake. At the mid-span surface both blade profiles are similar, however one can see the effect the blade compounded has generated. The compound lean will slightly increase the loss at the mid-span at the expense of reducing the loss at the near-wall regions. However overall the loss for the bowed blade is lower than the cylindrical blade.

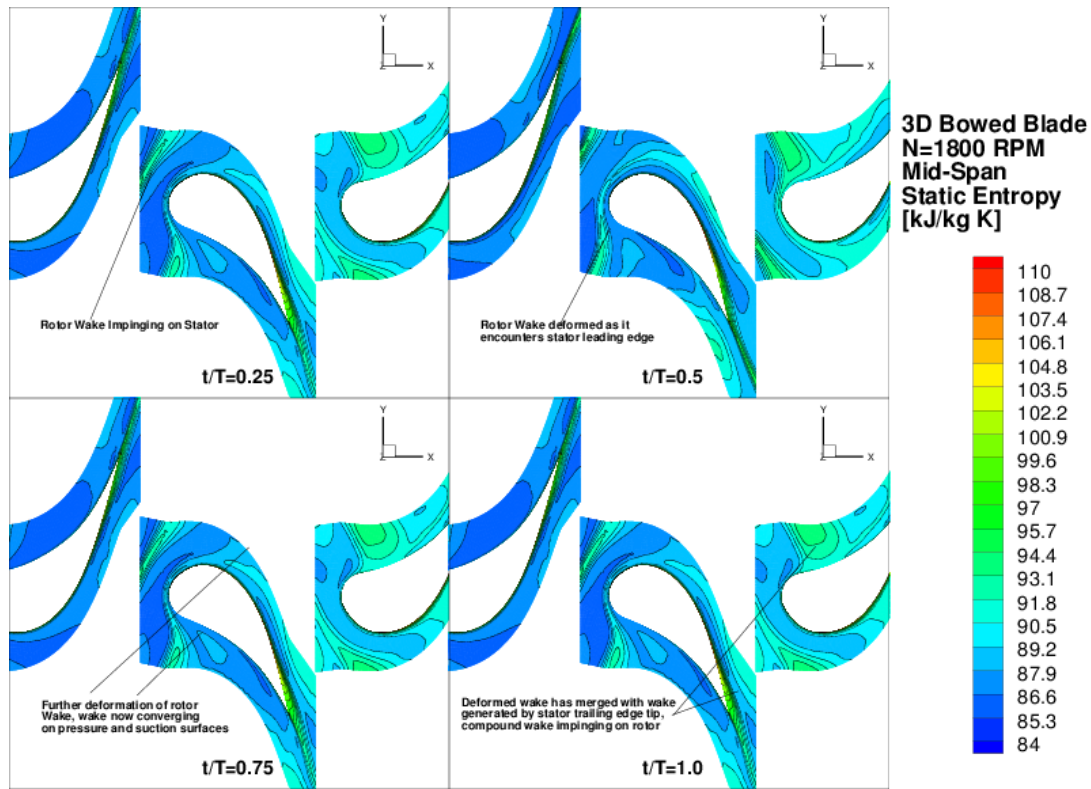


Figure 8.66: 3D bowed blade 1800 RPM mid-span entropy contours at various blade passage instants (rotation of rotor blades not shown).

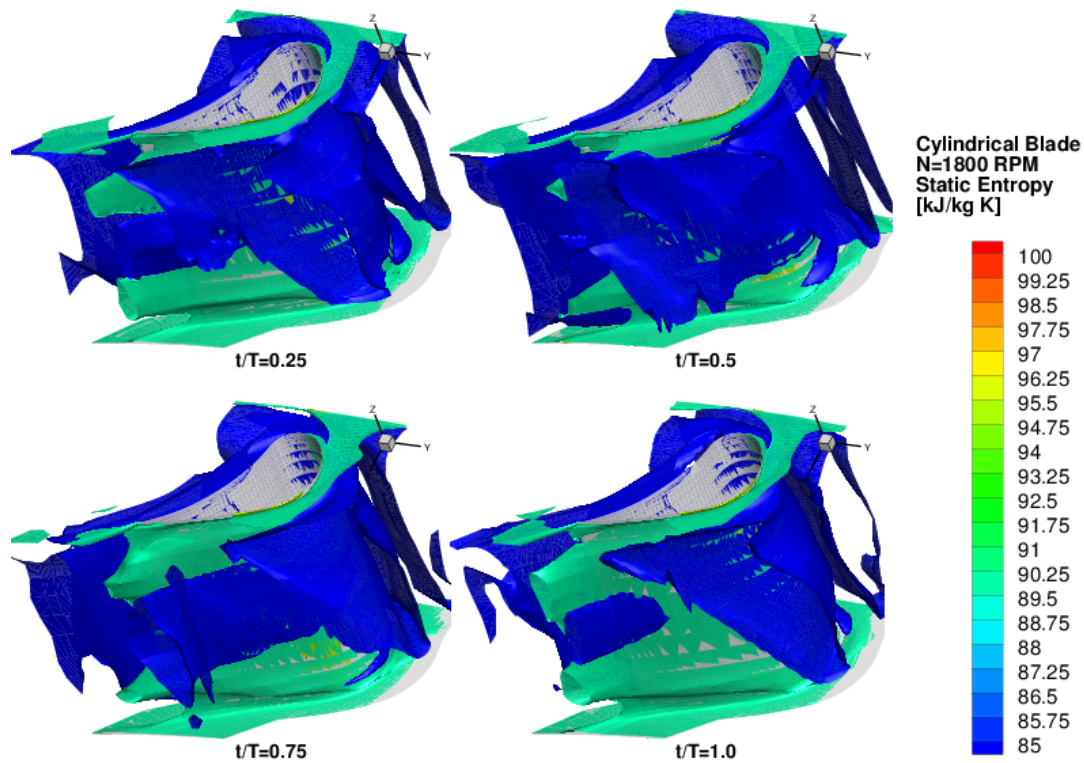


Figure 8.67: Cylindrical stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (suction surface view).

The wakes can be visualized in a 3D sense by creating isosurfaces of entropy as shown in figures 8.68 to 8.70, for both of two blade designs. Figure 8.68 shows a view of the suction side of the wake as it passes from the leading edge to the trailing edge, the loss value in the mid-span region is higher than the cylindrical blade case in figure 8.67. The pressure side portion of the leading edge wake can be observed for the cylindrical and 3D bowed blades in figures 8.69 and figure 8.70 respectively. The pressure side wake isosurface for the cylindrical blade shows lower entropy values however, it extends fully from hub-to-tip and covers a large area when compared to compound lean blade. More details can be seen in figures 8.71 and 8.72 which show various axial contours or slices of the entropy data for each of the four time steps through the wake passage period.

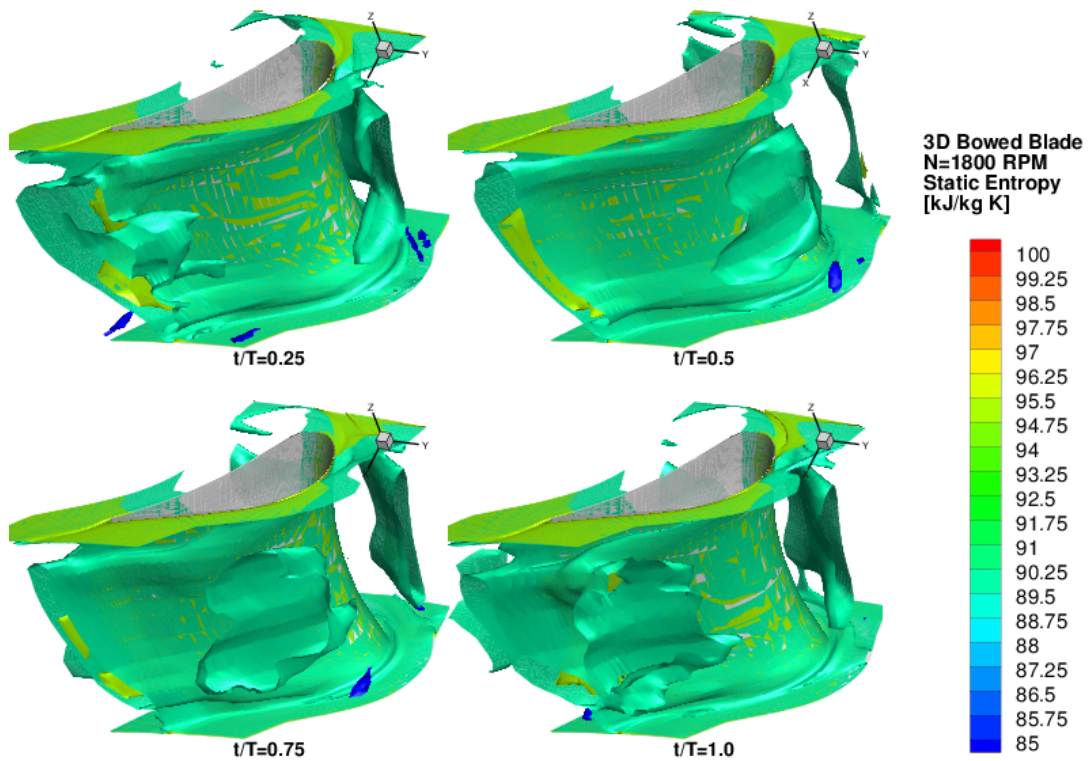


Figure 8.68: 3D bowed stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (suction surface view).

Table 8.4 shows a comparison between the predicted efficiencies using both RANS and URANS and the experimentally measured efficiency at the 1800 RPM point for both the cylindrical and 3D bowed series blades. The URANS provides a better efficiency prediction but the improvement with using the URANS is quite small when compared to the discrepancy to the experimentally measured efficiency. At the low speed point, the 3D bowed blade design still provides a slightly higher efficiency at this point.

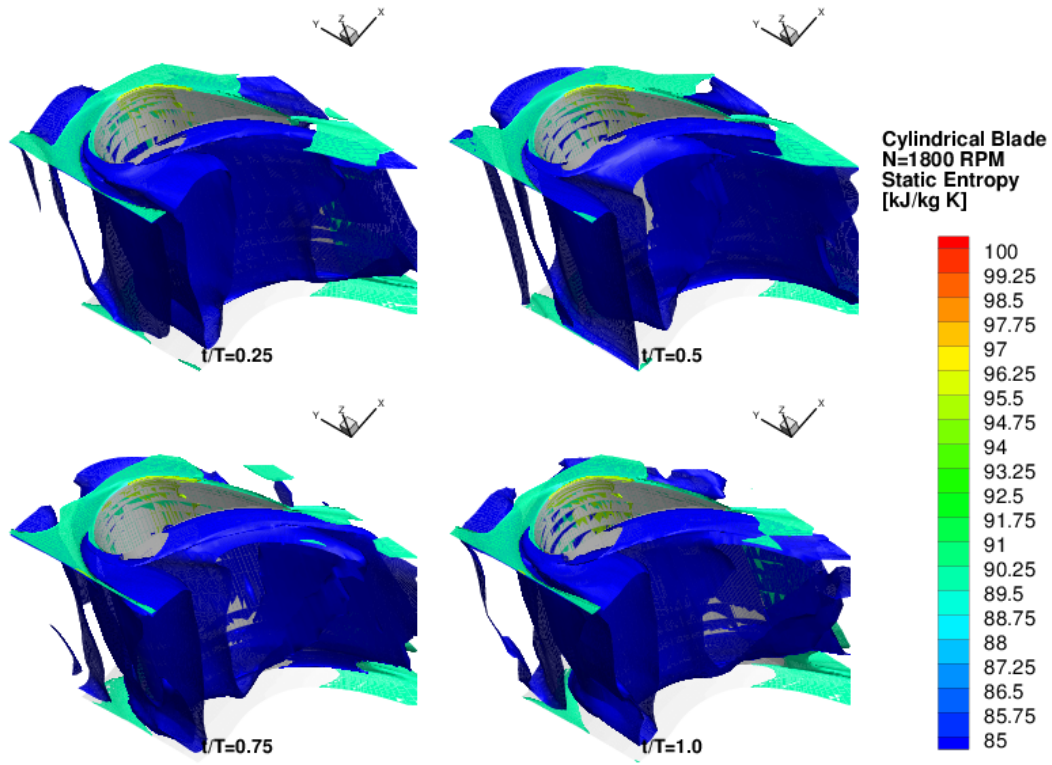


Figure 8.69: Cylindrical stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (pressure surface view).

Table 8.4: Total-to-static efficiency comparison for cylindrical and 3D bowed blades for the three-stage turbine 1800 RPM case.

Blade Series	Experimental η_{t-s}	RANS η_{t-s}	URANS η_{t-s}
Cylindrical	0.81262	0.83756	0.833
3D Bowed Blades	0.8170762	0.85174	0.84

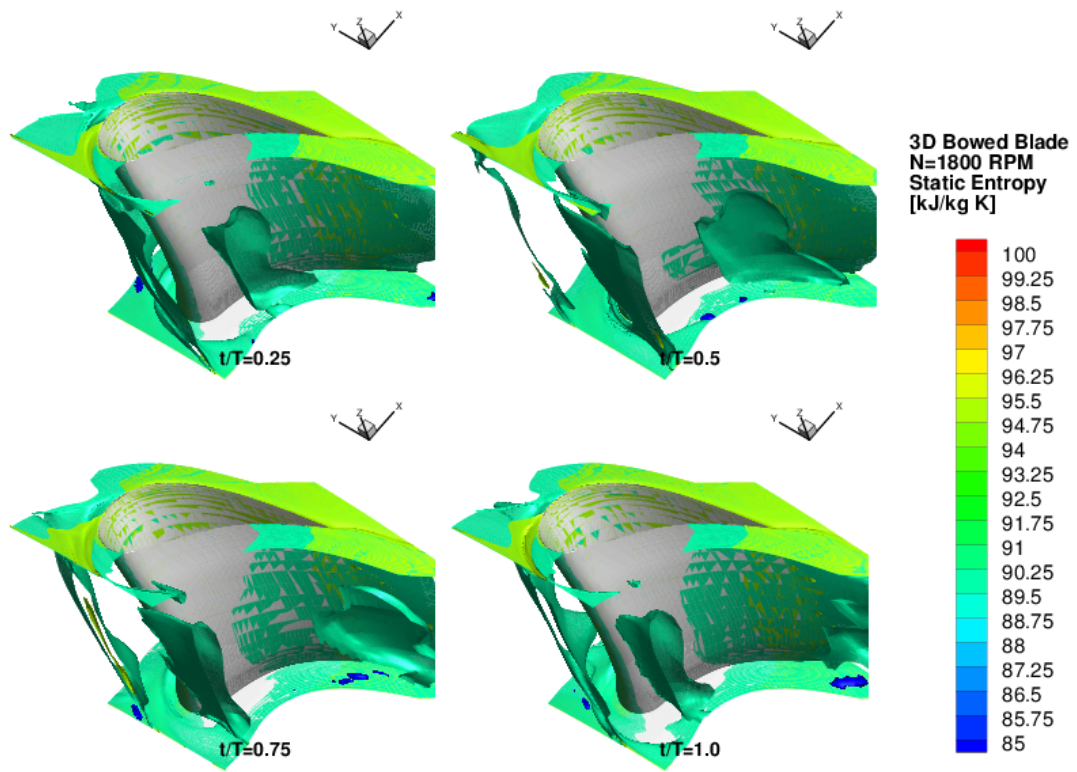


Figure 8.70: 3D bowed stator 2 blade 1800 RPM entropy iso-surfaces at various time steps (pressure surface view).

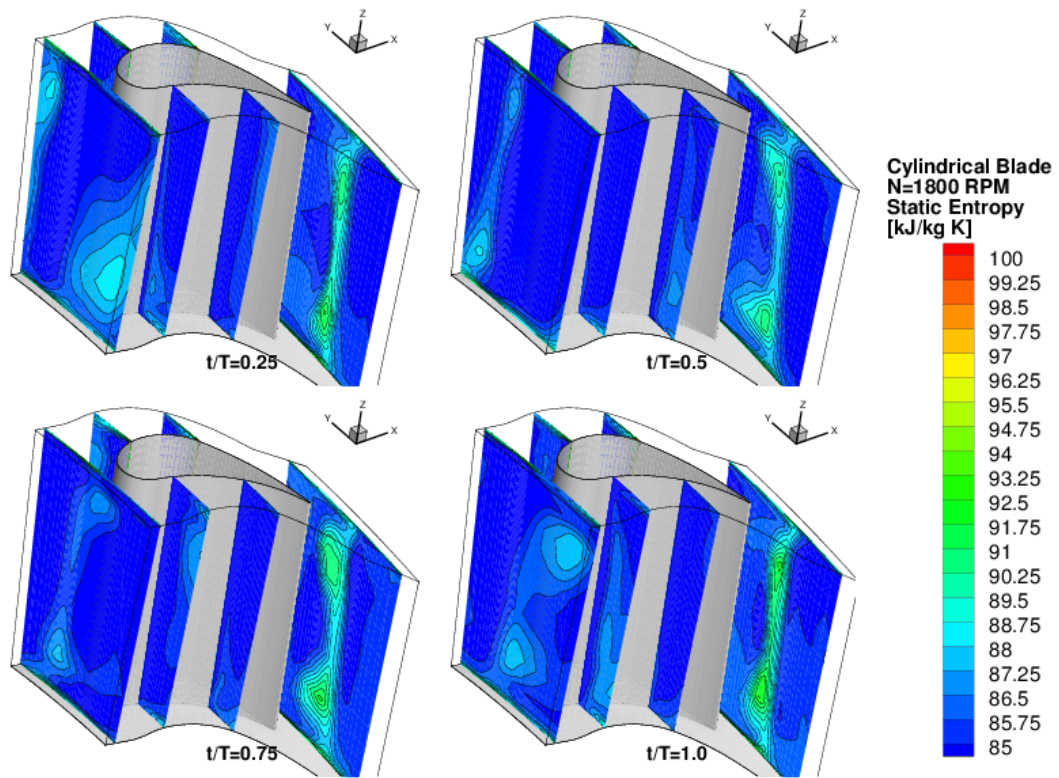


Figure 8.71: Cylindrical stator 2 blade 1800 RPM entropy contours at various axial positions, for various time steps

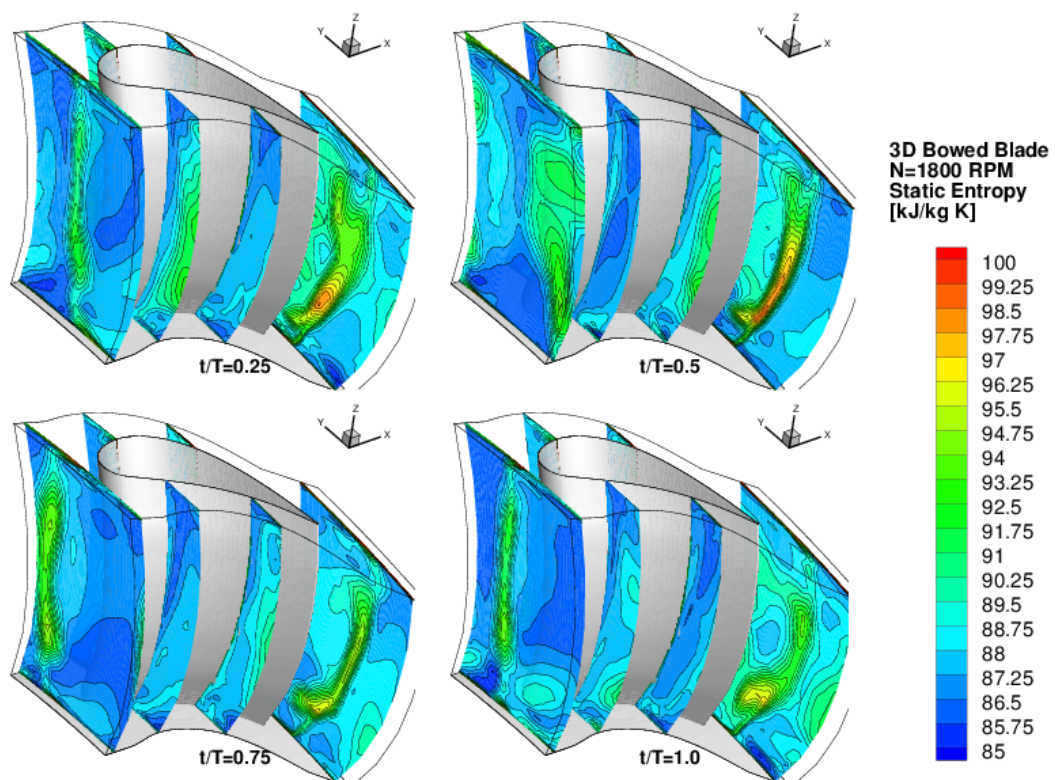


Figure 8.72: 3D bowed stator 2 blade 1800 RPM entropy contours at various axial positions, for various time steps.

9. DISCUSSION: 3D TURBOMACHINERY CFD

Over the past two chapters a CFD solver in both steady (RANS) and unsteady (URANS) modes, was used to analyze three different high pressure turbine blade designs. A 2-stage configuration, and then a three stage configuration which uses both cylindrical blade profiles in all its blade rows and positive compound lean stator and rotor blades. A comparison was made in all three cases to corresponding experimental data, as well as comparisons between various design points, which included high shaft speed, low shaft speed and in the case of the two stage blade the pressure ratio across the turbine was varied as well. CFD results in all cases shows significant discrepancies when compared to five-hole pressure probe based results. Namely, the pressure values at various interstage locations were significantly over-predicted, in some cases by up to 2 kPa, and in the case of velocities, near wall locations showed the largest disparity between CFD and experimental results where differences of about 20m/s (0.05 Mach) can be found. The overall interstage prediction resulted in a general over-prediction of performance at all operating points tested and especially at off design points (low pressure ratio and/or low shaft speed). In the case of the 2-stage turbine the difference between efficiency points observed was nearly 2.5 to 3 points, whereas for the three stage designs the solver fared much better, still over-predicting the total-to-static efficiency, but by a maximum of 2 efficiency points.

Looking at the flow patterns at each stage, it becomes very clear that CFD in general, and specifically for the RANS mode of simulation the wake profiles predicted by the solver are much more diminished when compared to the experimental data, which would generally explain why the efficiency predicted by the RANS solver is higher than the actual measurements. The author postulated that the RANS solver is burdened by a large handicap through the use of mixing planes between each blade row. The mixing planes will in effect wash out any upstream wake via circumferential averaging and then reimpose this average or 'mixed out' profile on the subsequent blade row as inlet boundary condition. This

process generates a mixing related entropy increase. In order to abate that, URANS cases were run, which fully coupled blade rows with a sliding mesh interface that requires an equal area match. The temporally and circumferentially averaged URANS results showed slight improvement however a significant performance discrepancy remained between experimental results and the two Navier-Stokes solvers. The improvement in URANS may have been hampered by the need to reduce mesh size and discretization as well as modifying the blade numbering to maintain a one to one area interface between stages. In a paper by Dunn et al. [7], changing the blade number to facilitate URANS cases resulted in slight differences in the predicted unsteadiness in the frequency domain, when compared to experimental results. However, useful information can be still be extracted out of the unsteady runs in terms of observing wake patterns, and how wakes are distorted as they traverse axially through the machine, and lastly, whether accounting for unsteady effects have further improved the efficiency predictions. In this case accounting for unsteadiness presented a small improvement in the predicted average efficiency or loss.

9.1 Modeling Errors

Modeling errors such as numerical errors are due to use of finite difference approximation to integrate the Navier-Stokes equations, other sources of numerical errors include the use of artificial viscosity or smoothing to stabilize various solver methodologies. The use of a linear interpolation across grid points presents another significant approximation, which will be amplified with larger distances (large mesh elements), usually the errors is on the order of Δx^2 where Δx is the grid spacing. To reduce this impact grids need to be refined as much as possible in regions which could contain large gradients or changes in second order derivatives, an example is the leading and trailing edge of a blade. Numerical errors due to grid spacing behave like an extra source of viscosity or numerical diffusion, and will as a result create extra unrealistic entropy, which can be convected downstream, and therefore influencing the solution further.

9.2 Mixing Planes

In order to enable a RANS simulation of a multistage turbine, mixing planes as described by Denton [4] are needed to mix out the upstream wake dominated flow to a pitchwise uniform enthalpy and entropy. The mixed out flow is then the inlet flow to the blade row downstream of the mixing plane. The mixing, or averaging creates an increase in entropy (loss in efficiency) at the interface planes. Unsteady interactions of the upstream wake with the downstream blade rows is significant in the case of transonic flow with shock structures, and thus the use of such mixing planes becomes contentious, however in all cases run in this study the flows were all subsonic and overall, the difference between URANS and RANS results was very small suggesting the URANS does not provide an advantage at the added computational costs incurred.

The mixing plane model used in the solver CFX, will circumferentially average, k , ω , ε , continuity, and enthalpy at the mixing planes. The mixed out or averaged values are then mapped onto the downstream 'patch', this is shown in figures 9.1 to 9.5. This mixing plane will remove any of the circumferentially based effects, such as wakes from the upstream blade row, postulating that across this interface, all these parameters will mix out or diffuse. This assumption will introduce an increased level of uncertainty in the RANS predicted values. Moreover, this assumes that this 'averaging' will be equal to the physical mixing that occurs between multiple frames of reference blade rows. In order to avoid the use of mixing planes a sliding mesh with a URANS solution is needed, however, at a much more computational cost.

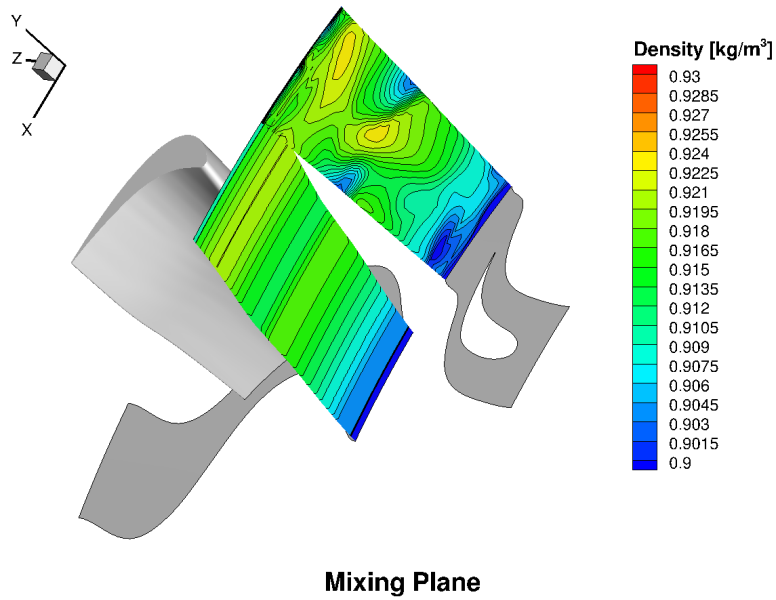


Figure 9.1: Mixing plane averaging applied to the density at the interface between the two-stage design's stator 2 and rotor 2.

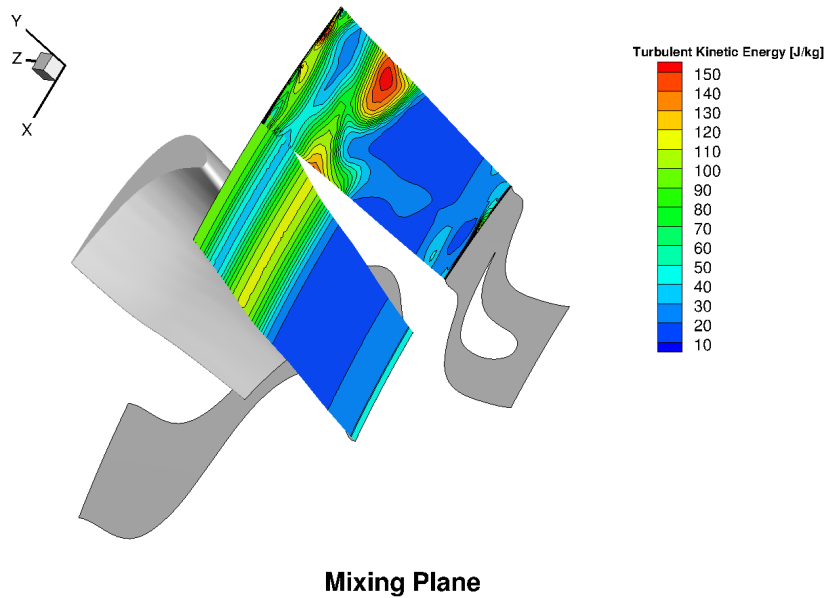


Figure 9.2: Mixing plane averaging applied to the turbulence kinetic energy at the interface between the two-stage design's stator 2 and rotor 2.

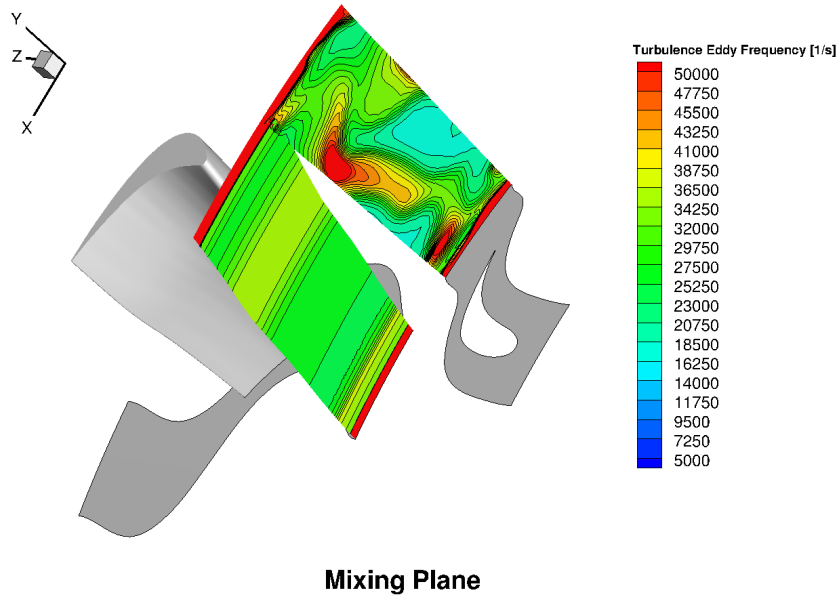


Figure 9.3: Mixing plane averaging applied to the turbulence eddy frequency at the interface between the two-stage design's stator 2 and rotor 2.

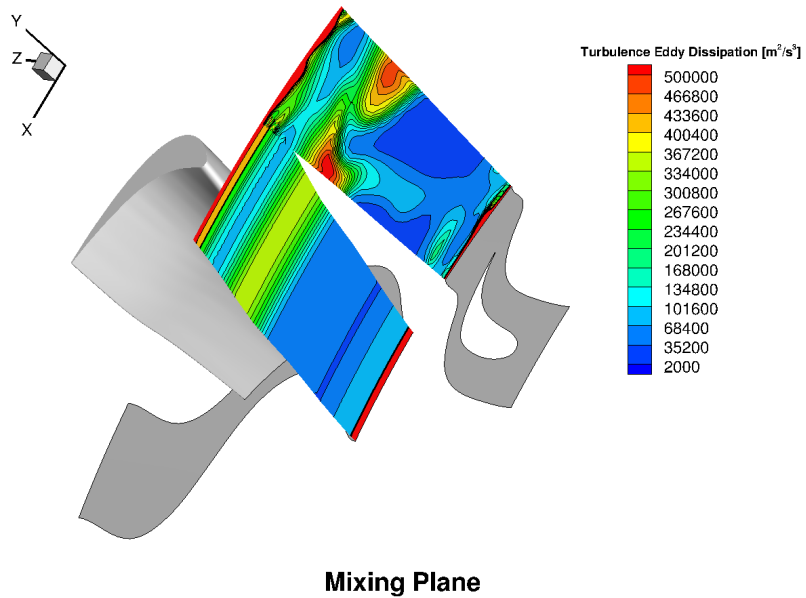


Figure 9.4: Mixing plane averaging applied to the turbulence eddy dissipation at the interface between the two-stage design's stator 2 and rotor 2.

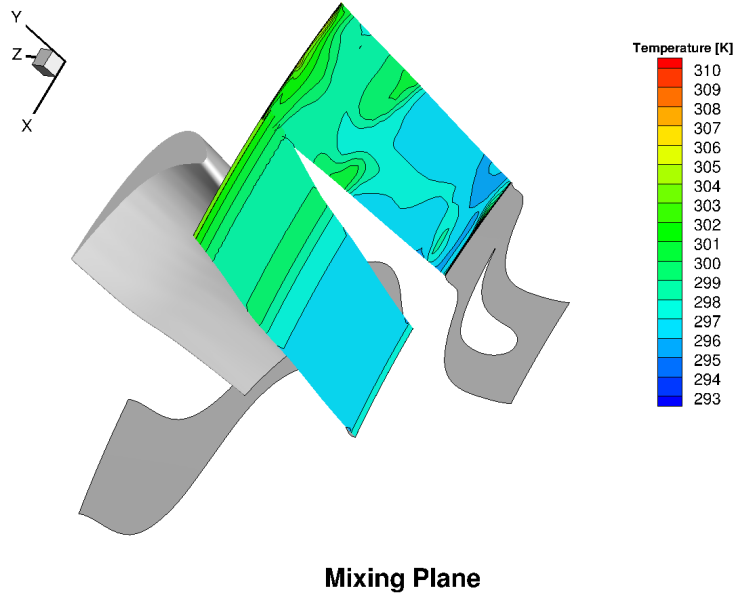


Figure 9.5: Mixing plane averaging applied to the temperature (enthalpy) at the interface between the two-Stage design's stator 2 and rotor 2.

9.3 Turbulence Modeling

Turbulence is a highly random and chaotic process, and to be fully described via modeling a grid discretization at an order of magnitude of the Kolmogorov length scale would be required, in other words solving or integrating down to the smallest scale vortices. This is done in direct numerical simulations (DNS) and to some extent in Large Eddy Simulations (LES) where a filtering function is used delineate a sub-grid scale and any sub-grid scale simulations will need a model, whereas any length scales about the sub-grid scale would be fully solved via DNS. Such DNS and LES computations are expensive, requiring extensive computational resources, and in the current turbomachinery design environment such computations cannot be done at an efficient manner for turbomachinery applications. Therefore, turbulence modeling is required to model the effect of turbulence, as stated by

Denton[4], in the case of fully turbulent and attached flows, the models should have been calibrated to give good flow predictions provided a appropriate discretization is used. However, in the case of separation or separating flows, most turbulence models have difficulty correctly predicting separating flow physics. Separating flows are unsteady in nature and as stated by Denton [4] flows at the edge of separation are likely to be affected more by the large length scale unsteadiness and not by the small scale turbulence that are modeled by various turbulence models. The Menter SST model used in all of the simulations in this study worked well overall especially at design points that have lower incidence and therefore less potential for flow separation, however at off-design cases, the flow predictions were found to over-predict the blade performance and under-predict the loss when compared to the experimental data. This suggests that any separation occurring at the high off-design cases due to large levels of incidence are under-predicted in magnitude.

9.4 Transition Modeling

Another aspect that can be a large part in predicting high pressure (HP) and low pressure (LP) turbine performance, is the use of a transition model to correctly predict transition from laminar to turbulent flow. The boundary layer scale has a large influence on entropy generation, most CFD codes can predict the growth pattern of a fully laminar or fully turbulent boundary layer well. However, transitional boundary layers are challenging because they depend on various factors such as inlet turbulence level, surface curvature and roughness, and pressure gradients. A separation bubble is the mechanism observed in transition at low turbulence levels (Reynolds numbers less than 5×10^6 , this bubble in most cases reattaches at location further downstream, and most transition models fail to predict the magnitude of the separation bubble and its reattachment due to wall curvature/damping effects. Inlet turbulence is a large factor in influencing the start of transition, higher turbulence will lead to an earlier transition, moreover, the highly turbulent and vortical nature of upstream blade wake passage on a downstream blade will heavily influence the transition point, such effects were extensively studied by Chakka and Schobeiri [53] and Halstead et al [52]. The situation is further convoluted by the near-wall effects and

the 3D nature of turbomachinery flows, were in some instances as stated by Denton [4] the boundary layer will revert to being laminar in nature. Predicting transition well can be crucial in attempting to predict the blade design's efficiency, for example in the case of the 9600 blade design the use of the transition model versus assuming the flow is fully turbulent results in a change of about 0.2 points of efficiency.

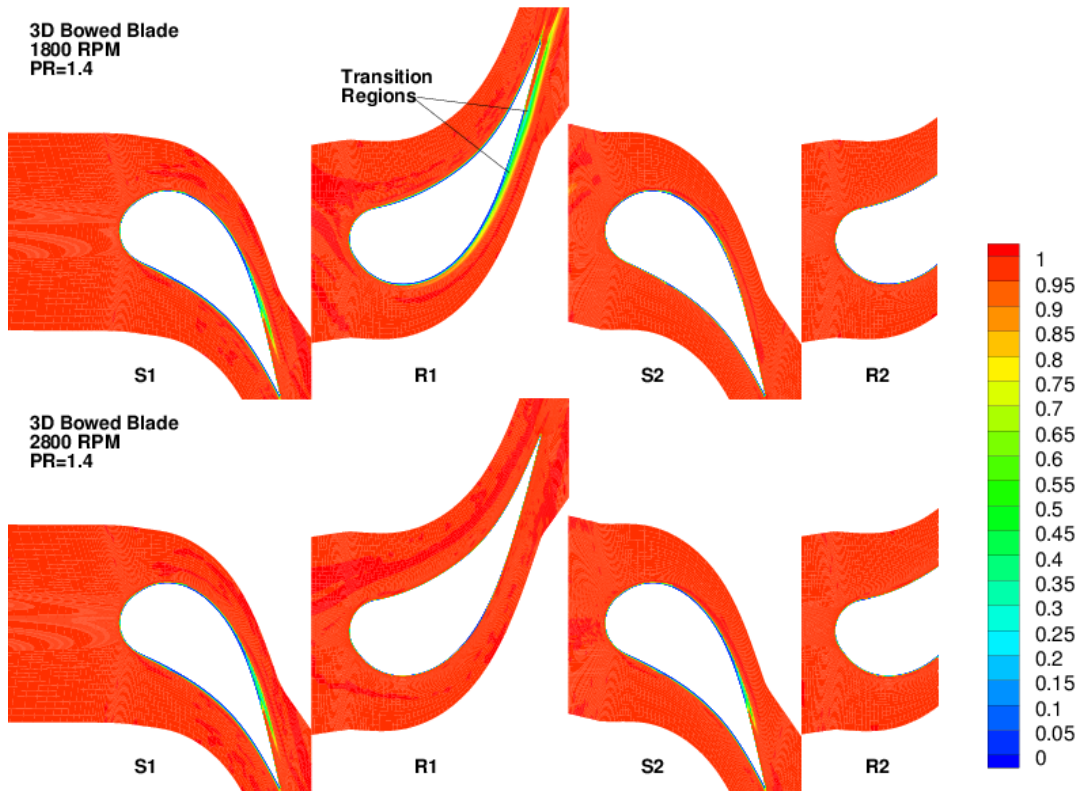


Figure 9.6: Contours of intermittency at the mid-span point for the three-stage bowed blade design for both the high and low speed case.

Figure 9.6 shows the predicted transition points for the 9600 blade design at high speed and low speed operating points. The first rotor blade is of interest where the different in-

cidence that is created between low speed and high speed points cause transition to trip much earlier on the suction surface of the blade as well as the pressure surface. In the high speed point the same blade row shows fully turbulent flow characteristics.

9.5 Experimental Considerations

There are many experimental considerations were comparing experimentally obtained data with CFD simulations of the blade design. Most critically, the boundary conditions, i.e. are the boundary conditions used in the CFD accurately describing the experimental data? In the simulations run in this study one major unknown was the inlet turbulence intensity. This parameter was not measured experimentally, and thus various values were tested in the CFD, ranging from 1% turbulence intensity to 5% showing little differences in the CFD data in that range. The experimental setup could not possibly imply a inlet turbulence intensity of greater than 5% due to the use of the baffle type heater at a significant upstream distance from the inlet plane, and through the extensive use of flow straighteners throughout the turbine facility at TPFL. Inlet turbulence intensity will have a significant impact on transition modeling as discussed above. In most cases the exit boundary conditions do not have a large impact on the inlet conditions and accurately gauging the inlet boundary conditions to be used with the CFD from a experimental run is shown to be difficult.

An important consideration are leakage flows due to seals or tip clearances, all configurations tested and simulated were fully shrouded (both stator and rotor rows) however a hypothetical tip clearance was introduced to the 2-stage design and its impact on the overall efficiency was significant. The analyst needs to consider the impact of leakage flows and tip clearances on the overall model and whether there is a need to fully model these effects using complex geometry or through a simpler use of source terms. This can be a significant source of error for CFD analysis of turbomachinery flows. Moreover, surface roughness, will play a large part in boundary layer growth, transition to turbulent flow, all experimental data was taken with newly machined blades and thus a roughness in the order of a few microns, however, with use the surface roughness will increase and bring

about a decrease of turbine performance with time. It will be difficult to accurately assess the surface roughness at various points in time moreover, surface roughness will still play a larger role especially in a transitional flow regime, where most surface roughness models will be deficient.

With all the CFD error sources discussed above, CFD remains an important academic and industry design tool that aids in further understanding turbomachinery flow physics, and from experience it has been shown that, CFD is a powerful tool used to compare between various design points and to provide further insight in the impact of various geometry and design characteristics on the overall design as it is taken from one operating point to the next. However as a pure a priori predictor of a design's performance, CFD still has much to overcome to fill that role. Turbulence model errors, transition model errors, numerical error sources as discussed will allow hamper CFD prediction of performance until the use of DNS becomes feasible as a design tool in the future. Denton [4] and Schobeiri et al. [2] confirm that CFD should be used on a comparative basis, i.e. comparing one point to another, rather than an absolute predictor of performance. Experimental testing will still be needed to improve quality of CFD and provide the basis of better turbulence models, transition models, and surface roughness among others, for the future advancement in turbomachinery based CFD.

10. LAMINAR TO TURBULENT TRANSITION MODELING

Fluid dynamics research has advanced extensively since the Osborne Reynolds experiment, however, transition from laminar to turbulent flow has yet to be fully understood and remains an important research topic in the field of fluid dynamics. The main challenges include the lack of a validated simulation of boundary layer transition. Despite the large amount of data that has been taken to study boundary layer transition there has not been a correlation developed that could reliably predict transition at various flow conditions and geometries.

One way to predict onset of transition is through using a boundary layer stability analysis, a detailed treatise of such analysis is presented in chapter 8 of [85]. The stability analysis results in the famous Orr-Sommerfeld charts. However, this method cannot be used in simulating bypass transition, and can only be used to predict the onset of transition and not for modeling the transition region or the fully developed turbulent flow region. Finally this method requires a converged steady-state laminar flow solution to the Navier-Stokes equations, which can be challenging with certain geometries.

The most popular way of simulating transition is via RANS based CFD codes with a low Reynolds number turbulence model. However it has been shown to be very difficult to provide accurate results of skin friction and heat transfer of a flow configuration that exhibits transition. This becomes even more difficult with the added influence of pressure gradients and free-stream turbulence intensities. Most modern transition models make use of transport equation for intermittency or kinetic energy. These models have to be augmented with empirical correlations to predict the onset and size or extent of the transition region.

10.1 Transition Models Summary

Transition models can be categorized into stability theory based models and non-stability theory based models, some of these models are briefly discussed here.

10.1.1 e^n Method

This method proposed by Smith and Gamberoni [100] and Van Ingen [101] is based on linear stability theory. It is based on three steps, first involves computing the laminar velocity and the temperature profiles at different stream wise locations for the given flow. Second, the amplification rates of the most destabilizing waves are computed for each profile. Last, the amplification rates are used to predict or obtain the transition location. The pitfalls of this method include the need to obtain an n factor for transition that is based on experimental data, and the use of linear stability theory with the assumption that the flow is locally parallel.

Most models do not have a means to predict onset, and rely on the e^n method or on empirical data obtained via experimental results.

10.1.2 Baldwin-Lomax Algebraic Turbulence Model

The transition region is modeled by switching of the turbulence model for the laminar region via zeroing out the eddy viscosity in the laminar flow region and then switching the eddy viscosity back on at the transition point.

10.1.3 Warren, Harris and Hassan One Equation Transition Model

This model uses the intermittency expression developed by Dhawan and Narisimha [102], which is defined as:

$$\Gamma = 1 - e^{\left[\frac{0.41(x-x_t)^2}{\lambda^2} \right]} \quad (10.1)$$

where x_t defines the location of transition onset. The model then uses a formula for the eddy viscosity length scale l_μ defined as:

$$l_\mu = (1 - \Gamma) [l_{TS} - l_{SM}] + \Gamma l_\mu^t \quad (10.2)$$

The expressions l_{TS} and l_{SM} are the length scale contributions from the first and second modes, and are based on experimental correlations. l_μ^t is the turbulent length scale.

10.1.4 Linear Combination Transition Model

Developed by Dey and Narasimha [103] and based on the concept that transition is a mix of laminar and turbulent flow regions, the contributions from each regime are applied based on the intermittency factor developed by Dhawan and Narasimha [102]. The model requires a complete laminar flow solution, which is then followed by the model run with turbulence activated, and with the knowledge of the starting point of transition.

Models that include the capability to predict onset location include:

10.1.5 Suzen and Huang Model

This model uses a transport equation for the intermittency factor, which includes two source terms from both the Steelant and Dick [104] model and the Cho and Chung model [105]. The intermittency is then used by multiplying it with the eddy viscosity value obtained with from the turbulence calculations. The SST turbulence model is used to calculate the eddy viscosity value. The onset of transition was determined by comparing the local Reynolds number with a transition onset Reynolds number Re_θ calculated using the correlation of Huang and Xiong [106]. Here Re_θ is a function of the free stream turbulent intensity and the acceleration parameter. The inlet Re_θ value is based on the inlet turbulence intensity entered by the user.

10.1.6 Walters and Leylek Model

The model proposed by Walters and Leylek [107] uses the concept that bypass transition is brought about by high amplitude stream wise fluctuations. The fluctuations are different from the turbulence fluctuations and Mayle and Schulz [108] proposed a second kinetic energy equation be used to describe the fluctuations. This kinetic energy is termed the laminar kinetic energy k_l . In the near wall regions the regular turbulent kinetic energy; which in this model is called k_T is split into small scale energy and large scale energy, where the small scale energy contributes to the production of laminar kinetic energy and the large scale energy contributes to the production of the laminar kinetic energy.

10.1.7 $\gamma - Re_\theta$ Model

This model is based on the work of Langtry et al. [81] and Menter et al. [80], and is based on vorticity Reynolds number, which is used as a local parameter. As the authors state in [80], the vorticity Reynolds number is directly proportional to the momentum thickness Reynolds number, therefore, the vorticity Reynolds number can be used to trigger laminar-to-turbulent transition. The model requires the solution of two transport equations for both the intermittency factor γ and the momentum thickness number Re_θ (greater details available in [80], but both transport equations are listed here for reference):

$$\frac{\partial(\rho\gamma)}{\partial t} + \frac{\partial(\rho U_i \gamma)}{\partial x_j} = P_{\gamma 1} - E_{\gamma 1} + P_{\gamma 2} - E_{\gamma 2} + \frac{\partial}{\partial x_j} \left[\left(\mu + \frac{\mu_t}{\sigma_f} \right) \frac{\partial \gamma}{\partial x_j} \right] \quad (10.3)$$

$$\frac{\partial(\rho \widetilde{Re}_{\theta t})}{dt} + \frac{\partial(\rho U_j \widetilde{Re}_{\theta t})}{\partial x_j} = P_{\theta t} + \frac{\partial}{\partial x_j} \left[\sigma_{\theta t} (\mu + \mu_t) \frac{\partial \widetilde{Re}_{\theta t}}{\partial x_j} \right] \quad (10.4)$$

The transition model of Menter [80] interacts with the Shear Stress Transport turbulence (SST) model via:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial(\rho U_j k)}{\partial x_j} = \widetilde{P}_k - \widetilde{D}_k + \frac{\partial}{\partial x_j} \left[(\mu + \sigma_k \mu_t) \frac{\partial k}{\partial x_j} \right] \quad (10.5)$$

where the production term \widetilde{P}_k is modified (only in the k-equation) as:

$$\widetilde{P}_k = \gamma P_k \quad (10.6)$$

where the original production term of the SST model [80] was defined as:

$$P_k = \min \left(\tau_{ij} \frac{\partial U_i}{\partial x_j} \right) \quad (10.7)$$

and the destruction term is modified as:

$$\widetilde{D}_k = \min (\max (\gamma_{eff}, 0.1) 1.0) D_k \quad (10.8)$$

The original destruction term of the SST model [80] was defined as:

$$D_k = \frac{9}{100} k \omega \quad (10.9)$$

The closure terms for the transition model are listed in detail in Langtry and Menter [109], the transition source terms are defined as:

$$P_{\gamma 1} = 2 F_{length} \rho D [\gamma F_{onset}]^{c_{\gamma 3}} \quad (10.10)$$

$$E_{\gamma 1} = P_{\gamma 1} \gamma \quad (10.11)$$

D is the magnitude of the deformation tensor. The empirical correlation F_{length} controls the length of the transition region. The destruction or relaminarization terms are defined as:

$$P_{\gamma 2} = (2 c_{\gamma 1}) \rho \Omega F_{turb} \quad (10.12)$$

$$E_{\gamma 2} = c_{\gamma 2} P_{\gamma 2} \gamma \quad (10.13)$$

The Ω term is the magnitude of the rotation tensor, and the onset is controlled using:

$$Re_{\nu} = \frac{\rho y^2 D}{\mu} \quad (10.14)$$

where y is the distance from the nearest wall.

$$R_T = \frac{\rho k}{\mu \omega} \quad (10.15)$$

$$F_{onset1} = \frac{Re_{\nu}}{2.193 Re_{\gamma c}} \quad (10.16)$$

$$F_{onset2} = \min \left(\max \left(F_{onset1}, F_{onset1}^4 \right), 2.0 \right) \quad (10.17)$$

$$F_{onset3} = \max \left(1 - \left(\frac{R_T}{2.5} \right)^3 \right) \quad (10.18)$$

Table 10.1: Intermittency gamma (γ) equation constants.

Constant	Value
$c_{\gamma 1}$	0.03
$c_{\gamma 2}$	50
$c_{\gamma 3}$	0.5
σ_y	1.0

$$F_{onset} = \max(F_{onset2} - F_{onset3}, 0) \quad (10.19)$$

$$F_{turb} = e^{\left(\frac{Re_T}{4}\right)^4} \quad (10.20)$$

The term, $Re_{\theta c}$ is the critical Reynolds number, when intermittency first starts to increase in the boundary layer. Both F_{length} and $Re_{\theta c}$ are function of the transport term $\tilde{Re}_{\theta t}$. The correlations for F_{length} and $Re_{\theta c}$ were not made public until the paper in [109]. The constants for the intermittency equation are listed in table 10.1 The closure terms for the momentum thickness transport equation are [109]:

$$P_{\theta t} = c_{\theta t} \frac{\rho}{t} \left(Re_{\theta t} - \tilde{Re}_{\theta t} \right) (1 - F_{\theta t}) \quad (10.21)$$

$$t = \frac{500\mu}{\rho U^2} \quad (10.22)$$

Here U is the nominal velocity magnitude.

$$F_{\theta t} = \min \left(\max \left(F_{wake} e^{-(y/\delta)^4}, 1.0 - \left(\frac{y - 1/50}{1 - 1/50} \right)^2 \right), 1.0 \right) \quad (10.23)$$

$$\theta_{BL} = \frac{\tilde{Re}_{\theta t} \mu}{\rho U} \quad (10.24)$$

$$\delta_{BL} = \frac{15}{2} \quad (10.25)$$

$$\delta = \frac{50\Omega y}{U} \delta_{BL} \quad (10.26)$$

$$Re_{\omega} = \frac{\rho \omega y^2}{\mu} \quad (10.27)$$

$$F_{wake} = e^{\left(\frac{Re_{\omega}}{1 \times 10^5}\right)^2} \quad (10.28)$$

The constants for the momentum thickness Reynolds number equation are listed in table 10.2

Table 10.2: Momentum thickness Reynolds number Re_{θ} equation constants.

Constant	Value
$c_{\theta t}$	0.03
$\sigma_{\theta t}$	2.0

10.1.8 Chakka and Schobeiri Model

This model is based on the curved plate and turbine cascade experiments by Chakka and Schobeiri [53], Schobeiri and Wright [110], Schobeiri and Chakka [111] and Schobeiri et al. [112]. The model introduces a relative intermittency term, that temporally relates between the maximum intermittency and minimum intermittency that is imparted onto a blade, due to a wake impingement (maximum intermittency or γ_{max} point) or a becalmed region (minimum intermittency or γ_{min} point). In addition to natural transition this model will account for wake induced transition as well, moreover, through experimentation of different reduced frequency values the relative intermittency term was shown to allow follow a Gaussian distribution as shown in [53] and figure 10.1. Figure 10.1a shows the points that correspond to the minimum and maximum intermittency γ values.

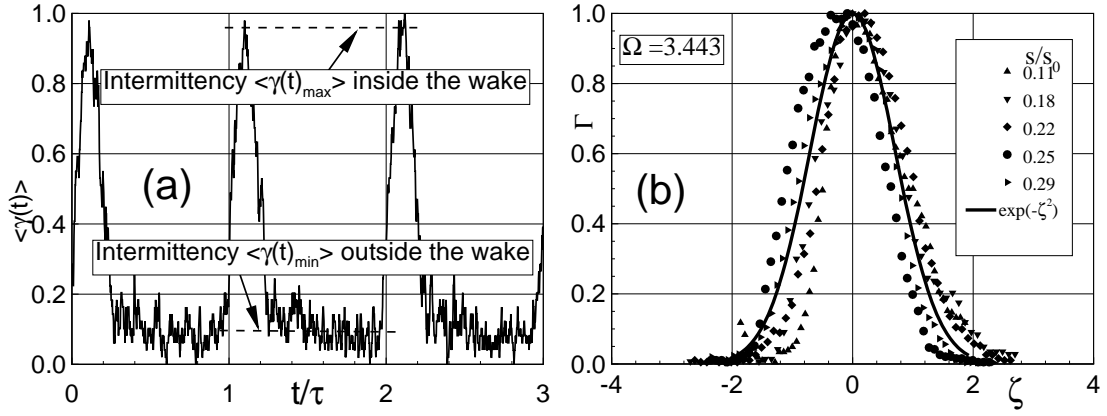


Figure 10.1: Relative intermittency Γ as a function of non-dimensionalized lateral coordinate for $\Omega = 3.443$, obtained from Chakka and Schobeiri [53].

This model will be discussed extensively in the next section 10.2, as an implementation of this model into a CFD solver is presented.

10.2 Experiments of Laminar to Turbulent Transition

In this section the experiments of Chakka and Schobeiri [53] are discussed in detail and simulated using two approaches. Firstly through the implementation of the model into the commercial code ANSYS CFX and then using the Menter [80] transition model. The experiments were performed on curved plate which mimics the pressure surface of a blade, as well as a blade cascade. A motor driven rod turbulence or wake generator was used at the plate/cascade inlet to simulate wake passage of an upstream rotor stage, and via the use of hot-wire instrumentation the boundary layer was examined in details, and the effects of the inlet turbulence or wake passage frequency was observed. A transition model was proposed and implemented into the boundary layer solver TEXSTAN [113].

The curved plate test section is shown in figure 10.2, the facility uses a centrifugal fan driven by a 111kW 3-phase constant speed AC motor sized to generate a Reynolds number

of up to 8.8×10^5 at the nozzle exit. Through the use of a honeycomb flow straightener and a series of screens a free-stream turbulence of 1.2 percent is observed at the test section inlet. A squirrel type wake generator is used to simulate blade wake passage at the exit of a turbomachinery rotor row. A frequency controlled electric motor is used to drive the squirrel cage wake generator and is run a counter-clockwise direction. As stated in Chakka and Schobeiri [53], the unsteady characteristics of the flow generated by the wake generator are measured by the unsteady parameter Ω , which is similar to the Strouhal number, and in turbomachinery terms termed the reduced frequency parameter. It is defined here by:

$$\Omega = \frac{\sigma}{\phi} = \frac{s_o}{s_r} \frac{U_{in}}{U_w} \quad (10.29)$$

Here σ is the ratio of the arc length of the plate s_o and the spacing between the rods s_r , and ϕ is the ratio of the inlet velocity U_{in} and the circumferential velocity of the wake generator U_w . Using hot wire anemometry measurements of the flow at various streamwise points were taken with wake generator run at various frequencies to mimic rotor blade wakes at different reduced frequencies impinging onto the curved plate which in this case behaves as the downstream stator row's pressure surface. The aim of this experiment was to observe the behavior of the boundary layer, and the transition from laminar to turbulent flow characteristics, whilst the wake passage occurs which impinges a cyclic series of belcamed region (inside the wake region) and a highly turbulent region (outside the work) onto a downstream vane, or as in the case of this experiment, a curved. This periodic process will certainly influence the natural transition from laminar to turbulent flow and bring about a so-called wake induced transition.

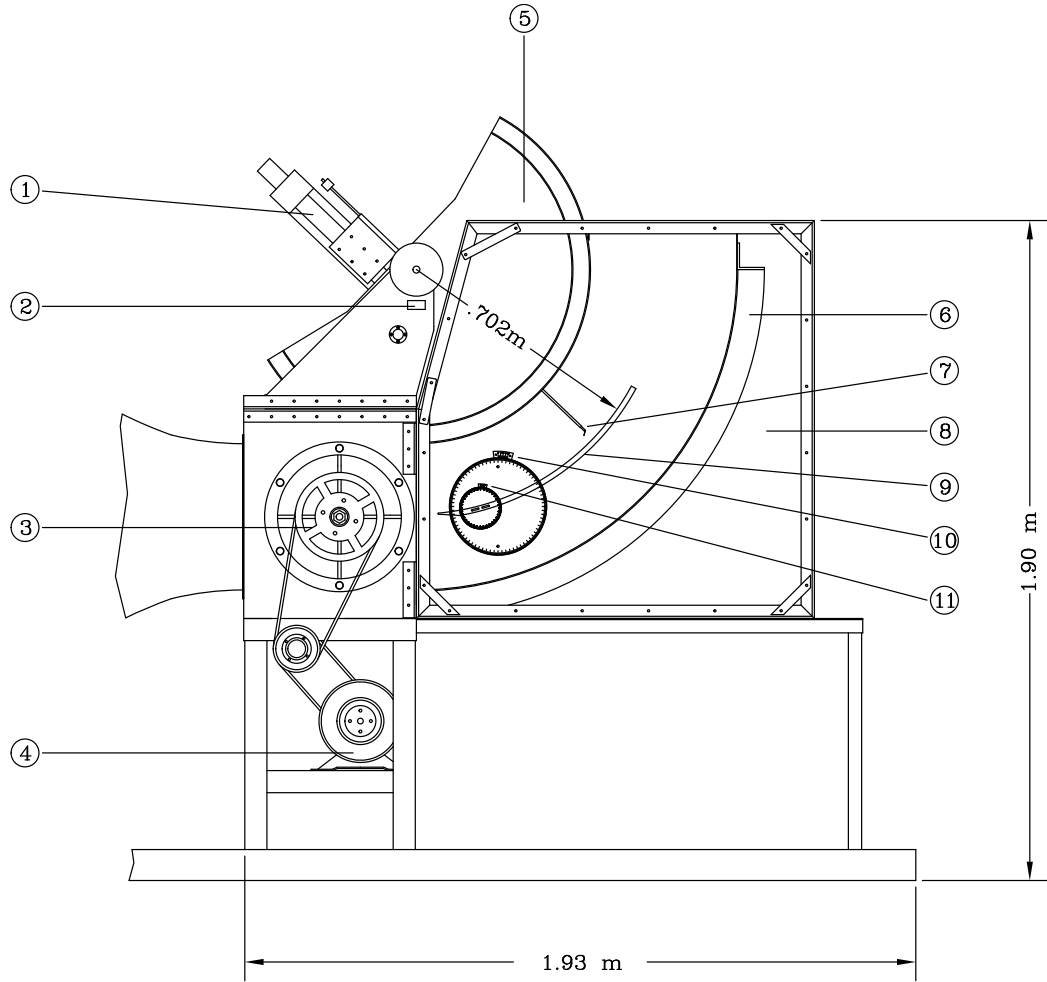


Figure 10.2: Curved plate test section of Chakka and Schobeiri [53] with 1. Traversing system, 2. Nozzle, 3. Wake Generator, 4. Electric motor, 5. Convex wall, 6. Concave wall, 7. Hot-wire probe, 8. Plexiglass wall, 9. Curved plate, 10. Small Vernier and 11. Large Vernier.

The experimental data was used to propose a transition model that accounts for this reduced frequency and offers the transition start and end Reynolds numbers for each reduced frequency. Chakka and Schobeiri [53] use a concept of relative intermittency, to quantitatively describe the wake influenced transition process. A wake defect can be described using a Gaussian function as shown in figure 10.1b, and the authors define a dimensionless

parameter ζ defined as:

$$\zeta = \frac{tU_w}{b} = \frac{ts_R}{\tau b} = \frac{\xi_2}{b} \quad (10.30)$$

where ζ is the time dependent length scale, and b , the wake width, is calculated using:

$$b = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \Gamma d\xi_2 \quad (10.31)$$

As detailed in [53] the relations above describe the passing time t of a wake impinging on the plate or blade row with a wake passage velocity U_w (belt velocity in the wake generator) and the intermittency width b . Relative intermittency can be defined as:

$$\Gamma = \frac{\gamma(t) - \gamma(t)_{min}}{\gamma(t)_{max} - \gamma(t)_{min}} \quad (10.32)$$

In equation (10.32) $\gamma(t)$ is the time-dependent ensemble-averaged intermittency function which describes the transitional nature of the unsteady boundary layer. The maximum intermittency $\gamma(t)_{max}$ is the time-dependent ensemble averaged intermittency value inside the wake vortical core, and the minimum intermittency $\gamma(t)_{min}$ is the time-dependent ensemble averaged intermittency value outside the wake vortical core. The maximum, minimum and average intermittency profiles as a function of the axial Reynolds number and at a specific reduced frequency of $\Omega = 1.033$ or using 3 rods in the wake generator are shown in figure 10.3

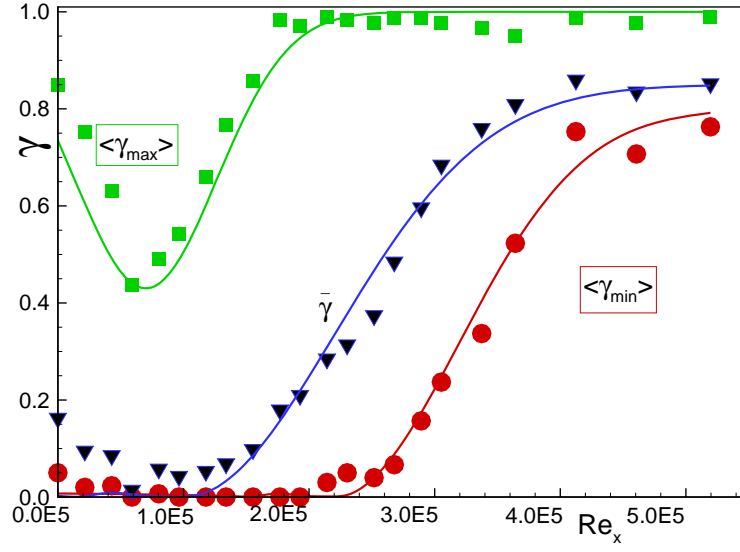


Figure 10.3: Maximum, minimum, and time-averaged intermittency distributions as a function of axial Reynolds number for $\Omega = 1.033$ (3 rods) at $y = 0.1$ mm, obtained from Chakka and Schobeiri [53].

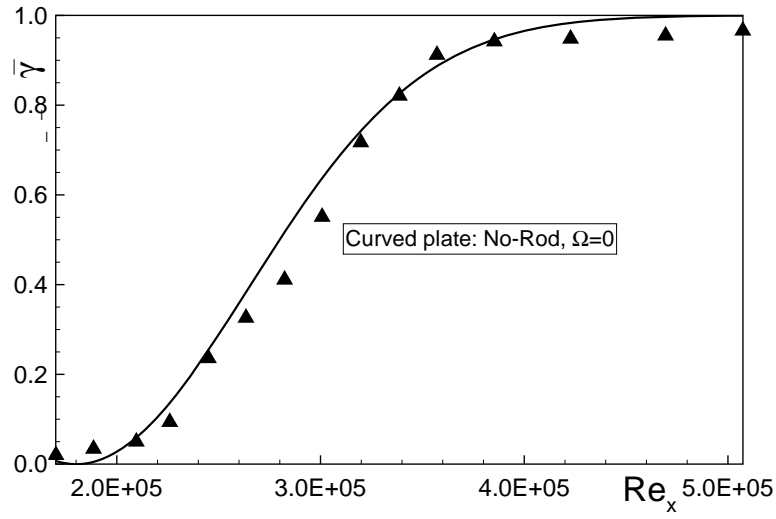


Figure 10.4: Averaged intermittency γ for the steady curved plate case $\Omega = 0$, obtained from Chakka and Schobeiri [53].

The steady case describes the natural transition process where the intermittency starts to increase at an axial Reynolds number of 2×10^5 and approaches the fully turbulent mode ($\gamma = 1$). The average intermittency $\bar{\gamma}$ for the steady curved plate case is shown in figure 10.4. With the unsteady case, two corresponding, and different intermittency states are present (albeit at various periodic phases). The maximum intermittency registers the point where the high turbulence intensity wake impinges on the plate surface, whilst the minimum intermittency represents the state where the wake is no longer present. The minimum profile follows the steady intermittency profile which starts from a value of zero and gradually increase to a value of 0.8, which the authors in [53] state is due to the transitional nature of the boundary layer. The maximum intermittency shows a different behavior, wherein it starts at a non-zero value due to the highly turbulent nature of the wake region, this potential to turbulence as expressed by the intermittency function is damped to a minimum value as the vortical wake is damped by the wall. As described by Chakka and Schobeiri [53], as the flow progresses axially a point is reached where the damping can no longer suppress the highly turbulent nature of the wake, thus increasing the intermittency where it approach a fully turbulent state. One can interpret this as the maximum intermittency shows the unsteady influence of the wake on altering the natural transition process, whilst the minimum intermittency shows the effect of natural transition without the influence of the unsteady wake. The combination of both results in a means to describe both the natural transition as well as the wake induced unsteady effects, resulting in a averaged intermittency that takes into account both effects.

Formulated into a model by [53] and reprinted below in equations (10.33) to (10.35), the model presents the minimum intermittency and the maximum intermittency, and a average intermittency that can be all be implemented into a Navier-Stokes based solver.

$$\gamma_{max} = 1 - c_1 e^{-\left[\frac{(Re-Re_s)}{(Re_e-Re_s)}\right]^2} \quad (10.33)$$

$$\gamma_{min} = c_2 \left(1 - e^{-\left[\frac{(Re-Re_s)}{(Re_e-Re_s)}\right]^2}\right) \quad (10.34)$$

$$\gamma_{avg} = c_4 \left(1 - c_3 e^{-\left[\frac{(Re - Re_s)}{(Re_e - Re_s)} \right]^2} \right) \quad (10.35)$$

The values for c_1, c_2, c_3 , and c_4 are a function of the reduced frequency parameter Ω . The values for the four constants are listed in table 10.3.

Table 10.3: Intermittency correlation constants, as listed in Chakka and Schobeiri [53].

Constant	$\Omega = 1.033$	$\Omega = 1.725$	$\Omega = 3.443$	$\Omega = 5.166$
c_1	0.57	0.22	0.50	0.35
c_2	0.80	0.85	0.86	0.88
c_3	1.00	0.82	0.80	0.80
c_4	0.85	0.92	0.92	0.94

10.3 Transition Model Implementation into CFD Solver

The model was implemented into the commercial solver CFX through the use of user based FORTRAN functions that are made into libraries, which are in turn accessible by the code at run-time. A function was generated to mimic the wake deficit nature of the inlet flow, another function was written provide the intermittency function. The intermittency subroutine, in transient mode would determine the relative intermittency (10.32) via the use of the Gaussian distribution approximation, the maximum and minimum intermittency is then calculated for each cell as function of the axial Reynolds number. In the steady mode, the relative intermittency is not needed a direct intermittency, a function of the axial Reynolds number and wall distance is subscribed to the code at runtime. All source code for the function is available in appendix B.1 of this thesis. The intermittency is implemented as follows into the $k - \omega$ based SST turbulence:

$$\frac{\partial}{\partial t} (\rho k) + \frac{\partial}{\partial x_j} (\rho u_j k) = \tilde{P}_k - \tilde{D}_k + \frac{\partial}{\partial x_j} \left[(\mu + \sigma_k \mu_t) \frac{\partial k}{\partial x_j} \right] \quad (10.36)$$

The intermittency is implemented in the SST model production term \tilde{P}_k and destruction term \tilde{D}_k , using:

$$\tilde{P}_k = \gamma P_k \quad (10.37)$$

where the original production term P_k is defined as:

$$P_k = \min \left(\tau_{ij} \frac{\partial U_i}{\partial x_j} \right) \quad (10.38)$$

and for the destruction term:

$$\tilde{D}_k = \min (\max (\gamma, 0.1), 1.0) D_k \quad (10.39)$$

The original D_k term is defined as:

$$D_k = \frac{9}{100} k \omega \quad (10.40)$$

Both production and destruction terms are unchanged from original SST model, however the intermittency γ is a multiplier to the production terms shown in (10.37), and a multiplier to the destruction limiter in (10.39).

10.3.1 Curved Plate Facility Grid

A numerical grid was generated to represent the plate geometry in figure 10.2. As shown by [81] it is important to maintain a wall Y^+ value of 1 or less to accurately capture the boundary layer characteristics, as a wall function is not used with the SST model.

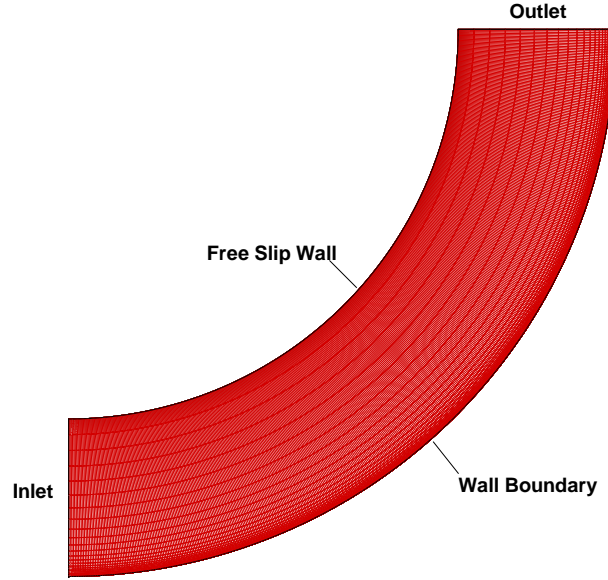


Figure 10.5: Numerical grid for plate facility transition simulations (coarse mesh with 38,000 elements shown).

The numerical grid used in the model is shown in figure 10.5 and incorporates the inlet and exit patch, a free-slip boundary condition wall and the plate surface represented by as fixed slip boundary condition wall. Three mesh configurations were used, coarse(38,000 elements), medium (600,000 elements) and finally fine (3,800,000 elements). The coarse mesh uses 77 nodes in the lateral direction, and 501 in the streamwise direction. As mentioned before care was made to maintain a Y^+ of less than 1 for all the grids, and in the case of the coarse grids this created really highly stretched elements away from the wall. Both steady and unsteady simulations were performed whilst using the model developed by Chakka and Schobeiri in [53].

10.4 Curved Plate Steady Results

Using figure 7 in [53] a Re_{start} of approximately 2.0×10^5 , Re_{start} was used as start value for transition, whilst a 3.8×10^5 value was used for the Re_{end} value. These Reynolds numbers are based on axial position, and not on the momentum thickness value as in the

case of the $\gamma - Re_\theta$ model. Through the use of the custom fortran based subroutines the a RANS solution of the curved plate was obtained using the three different grid densities, moreover, a case was run using the $\gamma - Re_\theta$ model of Menter as a means of comparison. Figure 10.6 shows the calculated momentum thickness at the curved plate wall as a function of the axial Reynolds number. As shown at the transition point an increase in the momentum thickness is observed. In the transition zone, the momentum thickness shows an irregular pattern. As the transition zone ends the momentum thickness starts to decrease in the turbulent zone due to plate curvature. Although the coarse grid (red) shows relatively good values of momentum thickness, it still has a tendency to over-predict when compared to the more dense grids. The denser grids have a greater streamline direction refinement, concluding that the solution shows little sensitivity to streamline refinement, however the near wall refinement is important for accuracy.

The Menter model shown in green matches the Chakka and Schobeiri model results well however, at the transition point the momentum thickness increases at a higher rate, and reaches a point that is nearly twice the value predicted by the Chakka and Schobeiri model. It is of interest to compare the momentum thickness Reynolds number to the axial Reynolds number, this is shown in figure 10.7. Based on the patterns shown in figure 10.7 the transition starts at approximately $Re_\theta = 200$, again the pattern observed by the Menter model shows the large increase in momentum thickness Reynolds number. This is due to momentum thickness Reynolds number equation used by the Menter model. The intermittency at the curved wall is shown in figure 10.8, the main difference is the intermittency value as used by the Menter model when compared to the model by Chakka and Schobeiri. As a intermittency transport equation is used in the Menter model, for a wall, the boundary condition specified for the intermittency is a Dirichlet boundary condition setting a zero gradient at the wall. In the case of the Menter model, this results in the lower values of γ at the wall as evidenced in figure 10.8. The Menter intermittency equation also specifies a intermittency value of 1 or fully turbulent at inflow patch. The Menter model shows an earlier start of transition at an axial Reynolds number of 1.7×10^5 as well as a

later end of the transition region.

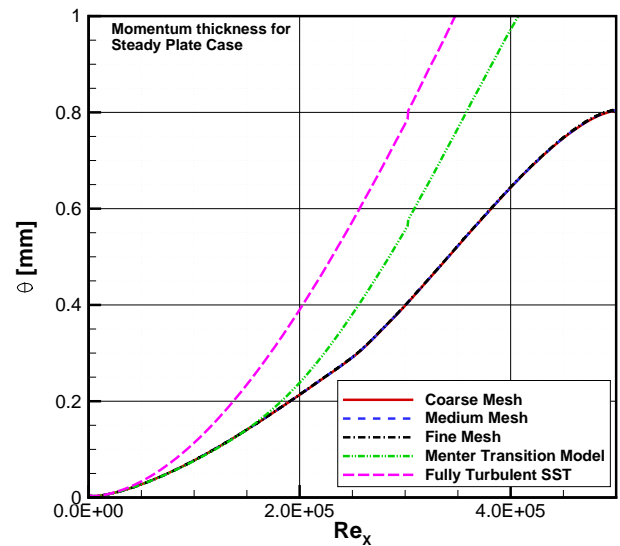


Figure 10.6: Momentum thickness θ as a function of axial Reynolds number at curved plate wall.

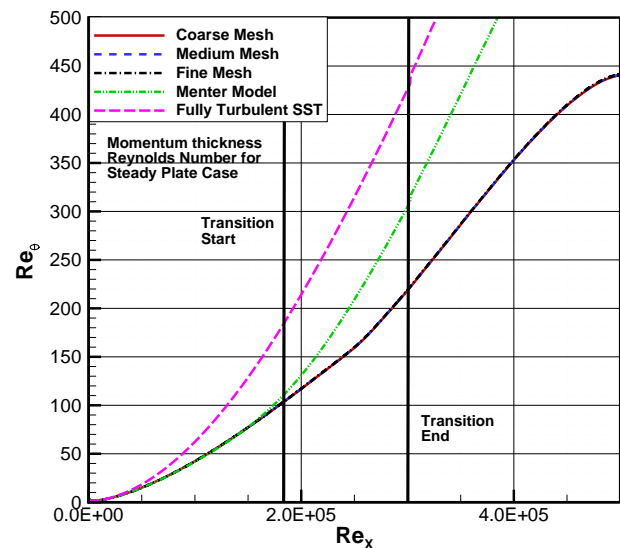


Figure 10.7: Reynolds number momentum thickness θ as a function of axial Reynolds number at the curved plate wall.

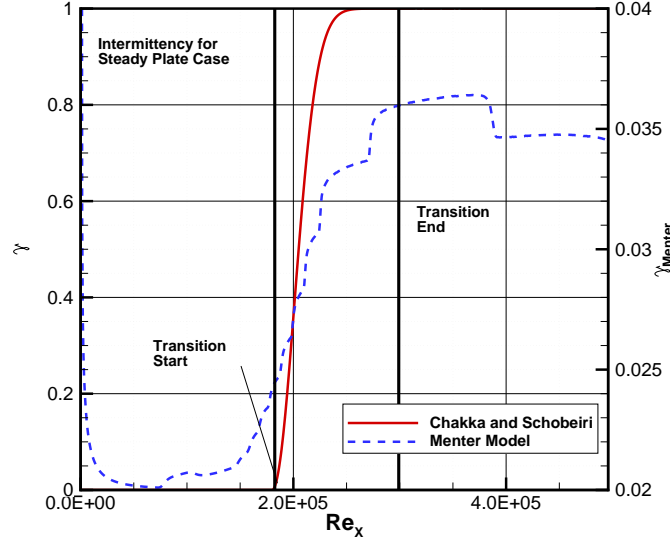


Figure 10.8: Turbulence intermittency γ as a function of axial Reynolds number at the curved plate wall.

In terms of heat transfer characteristics, a large difference is noted between the two models, as shown in figure 10.9 where although the intermittency begins at a value of 2×10^5 the increase in the heat transfer coefficient due to transition to turbulence occurs later at around $Re_X = 3 \times 10^5$. this trend is not exhibited by the Menter model, this is largely due to code numerics, and the damped nature of the production term P_k influencing the energy equation. Compared to heat transfer experimental results of Chakka and Schobeiri [53], although specifying the same start Reynolds number as measured in the steady case, the heat transfer transition occurs later axially than the what the experimental results show. The Menter model on the other hand shows a much earlier transition to turbulent flow when compared to the Schobeiri and Chakka model and the experimental data.

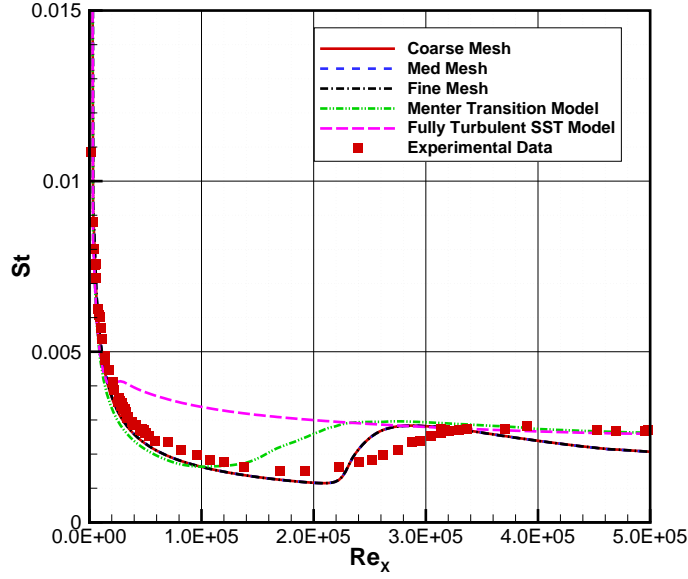


Figure 10.9: Stanton number as a function of axial Reynolds number at the curved plate wall.

However it is of note that the full effect of transition is observed at an axial Reynolds number of approximately 3×10^5 , although the model specifies a transition start at 2×10^5 , this 'lag' is an artifact of the solver. Using a FORTRAN function, the intermittency is applied as a multiplier to the production term as shown in (10.37), versus a more direct approach which as suggested by Chakka and Schobeiri in [53] and Schobeiri in [8], intermittency would become a multiplier of the turbulent viscosity term μ_t . It appears using the production term causes a the transition point to be predicted further downstream, and an abrupt increase in heat transfer or wall shear when the intermittency function reaches a value of 1, versus a more gradual increase as shown by the experimental data.

The data especially using the Chakka and Schobeiri model compares well to experimental data in [53] and provides an advantage over the Menter based transition model which shows a much earlier transition point. The Menter model is also highly sensitive to inlet turbulence boundary condition and it could be calibrated to match the data using the

Chakka and Schobeiri model via tuning the inlet turbulence, however it is best to show that in this case its predictive capability of the transition point at an inlet turbulence similar to the experiment, predicts a transition point that is slightly upstream of experimental data and the Chakka and Schobeiri model. Figure 10.10 highlights the difference between the intermittency function as used by Chakka and Schobeiri and the Menter model, its effect on velocity, and primarily boundary layer is illustrated further in figure 10.11. The results from CFD based on the model in [53] suggest that transition generally occurs at a momentum thickness Reynolds number Re_θ that is equal to approximately 200, however running the Menter model with this value as a fixed onset point results in a much earlier transition point as shown in figure 10.10.

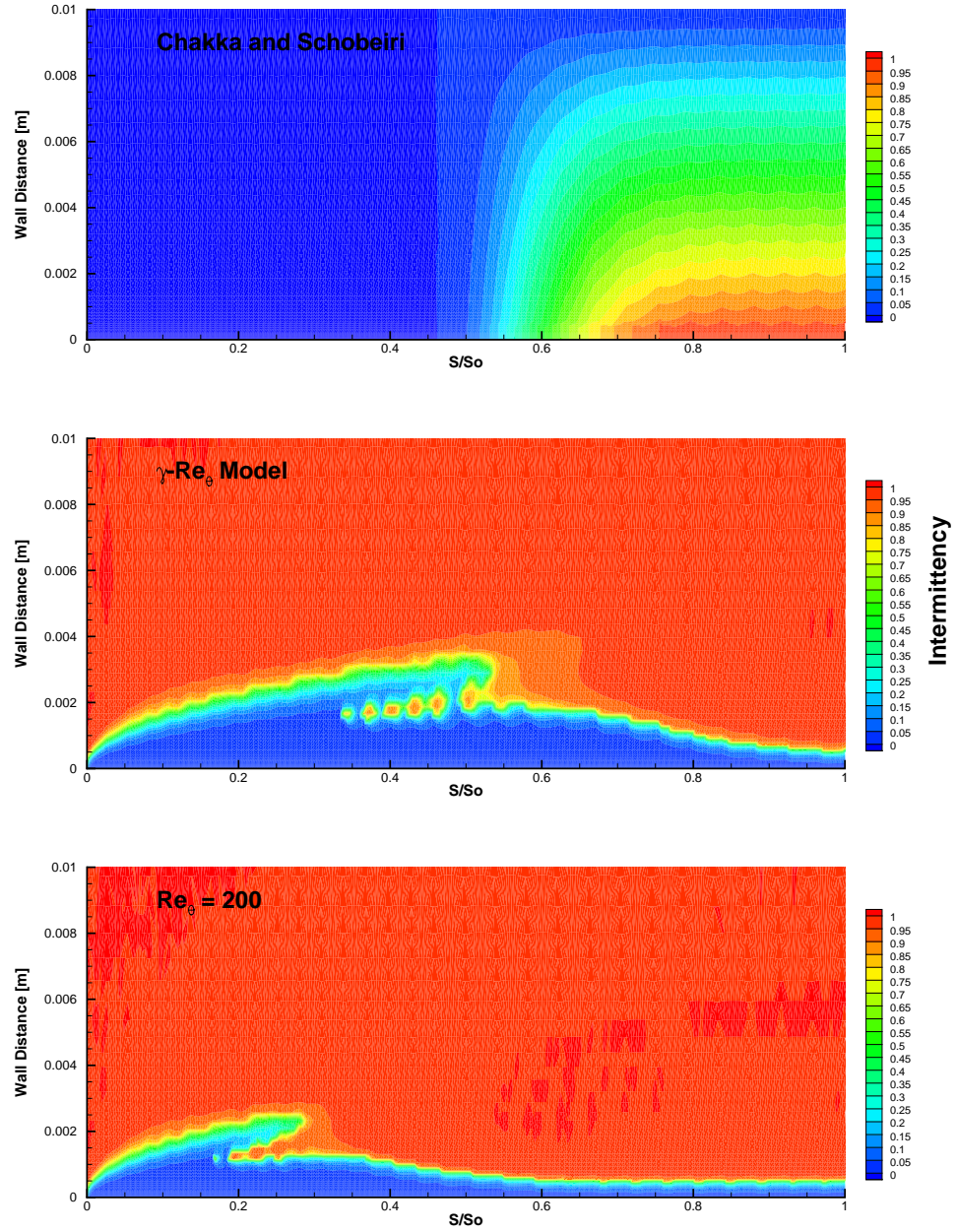


Figure 10.10: Contours of intermittency γ for steady curved plate case.

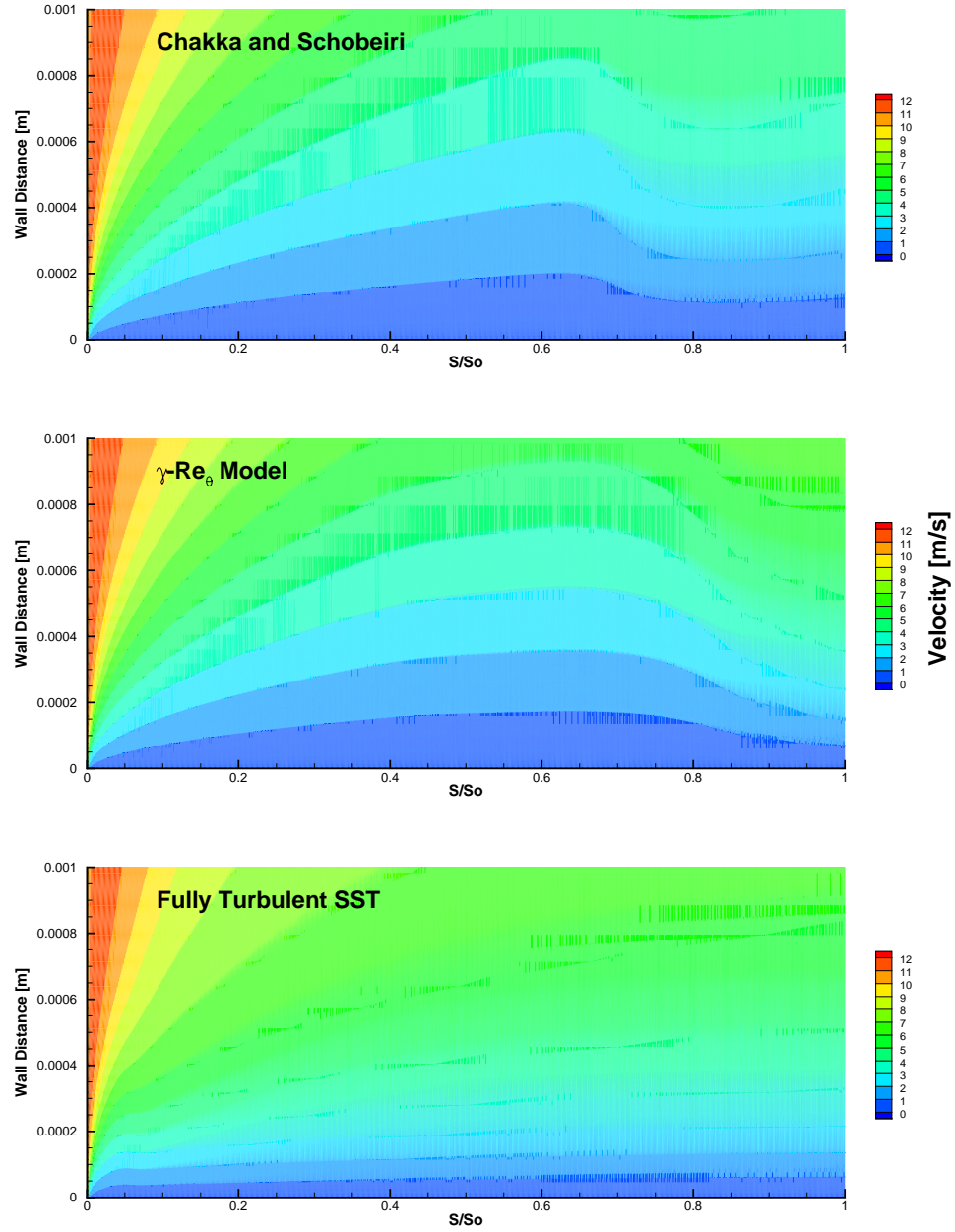


Figure 10.11: Contours of velocity for steady case using different transition models for curved plate case.

10.5 Curved Plate Unsteady Results

The curved plate was also run with a URANS solver, where the relative intermittency term Γ was employed to determine the intermittency as a function of time or of wake position, whereby a crucial balance between the maximum intermittency, achieved when the wake with a highly turbulent vortical core impinges on plate, and the minimum intermittency which occurs when the becalmed region of the wake is incident upon the plate. The indexing was modeled using a customized fortran function in CFX where it was related to the inlet velocity periodicity. When the high vortical core impinges on the plate the relative intermittency Γ was set to 1. Conversely when the becalmed region was impinging on the plate a relative intermittency of zero was used. Three cases of different reduced frequency were simulated and are listed below in table 10.4, all three cases used a time step of 5×10^{-4} seconds and were run to a time of 1 second with all time steps saved.

All cases run in unsteady mode are listed in table 10.4 which included different reduced

Table 10.4: Unsteady plate boundary conditions.

Case	Inlet Velocity[m/s]	Lateral Velocity[m/s]	Rod Spacing[m]	Reduced Frequency Ω
1	10.6	1.88	0.01571	0.78
2	10.6	3.76	0.0629	3.89
3	10.6	8	0.0629	8.28

frequency values that are similar to data taken by Chakka and Schobeiri [53], moreover the start and end points for transition as a function of reduced frequency Ω were based on data in [53]. Figures 10.12 and 10.13 show the resultant intermittency values at various times for a high and low reduced frequency case respectively. As noted the higher Ω value

results in an earlier transition point, whereas a lower reduced frequency results in a later transition point. This is further shown in the temporally averaged intermittency function shown in 10.16. The wall Stanton numbers at various time phases are shown in figures 10.14 and 10.15 for the high and low reduced frequency values respectively. It is observed with great interest that at higher reduced frequencies, the wake impinges onto the plate at a higher angle, and inherently causes transition to trip much earlier, this is observed very clearly in the Stanton number contours where higher heat transfer is occurring at a more upstream point in the plate when compared to lower values of reduced frequency. Moreover, as shown in the temporally averaged wall shear values 10.17, the lower reduced frequency value results in a slightly high wall shear, and a distinct late transition point, whereas the higher reduced frequencies results in an lower overall wall shear across the plate wall.

Figure 10.18 shows the change in the intermittency contour at various time phases, where the, highly vortical wake impinges on to the plate and moves downstream along the wall surface as the next wake emerges. The resultant near wall velocity contours at various time phases is illustrated in 10.19.

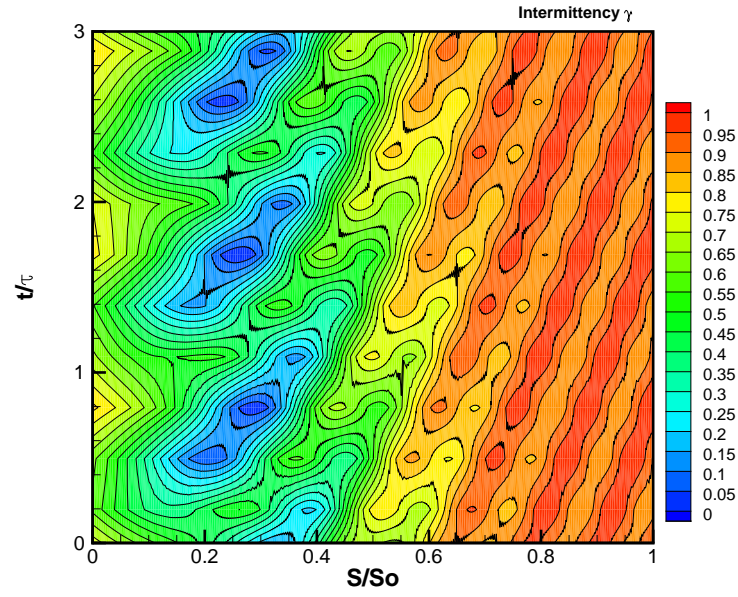


Figure 10.12: Unsteady intermittency for $\Omega = 3.78$.

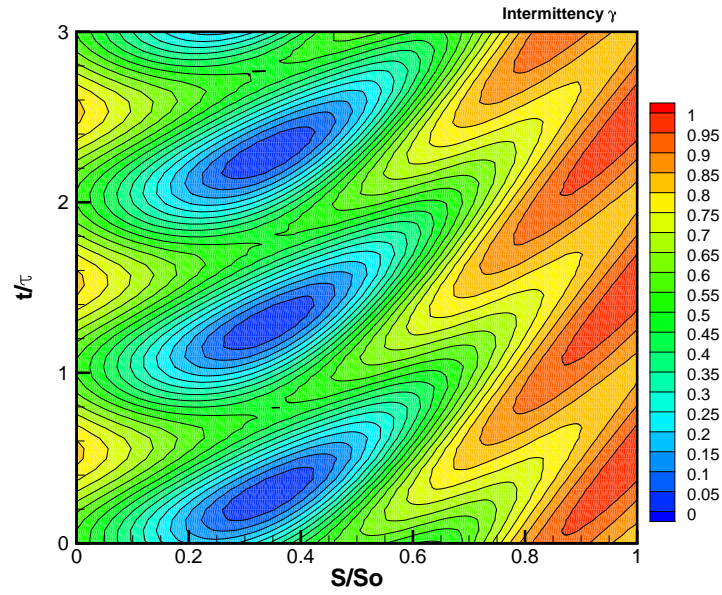


Figure 10.13: Unsteady intermittency for $\Omega = 0.77$.

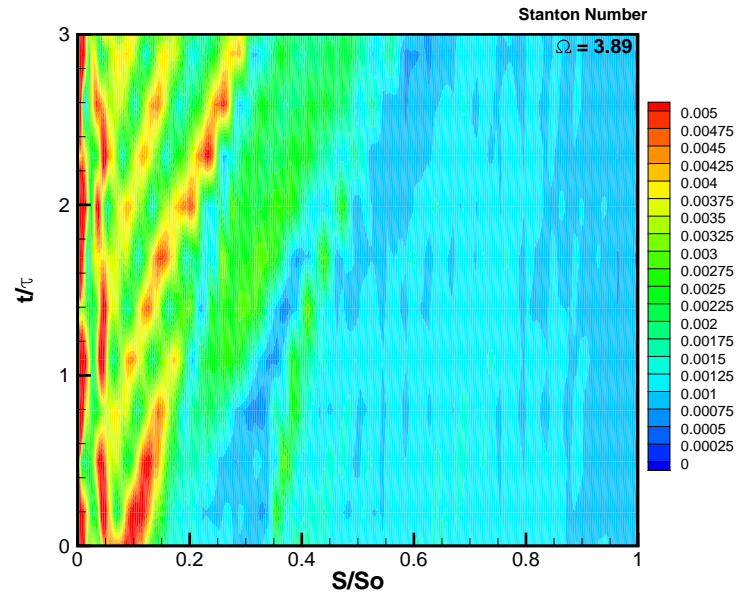


Figure 10.14: Unsteady Stanton number for $\Omega = 3.78$.

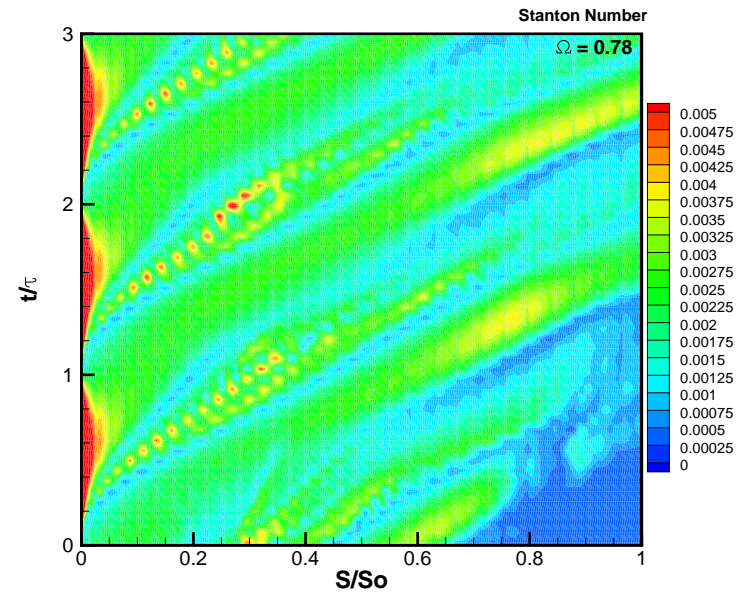


Figure 10.15: Unsteady Stanton number for $\Omega = 0.77$.

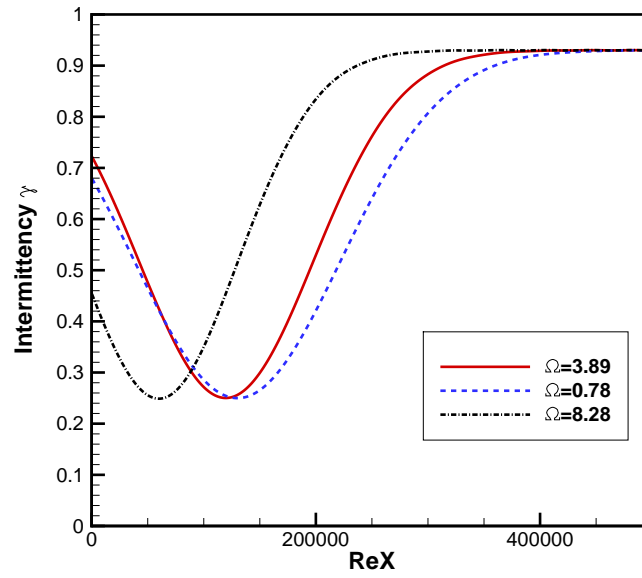


Figure 10.16: Temporally averaged intermittency.

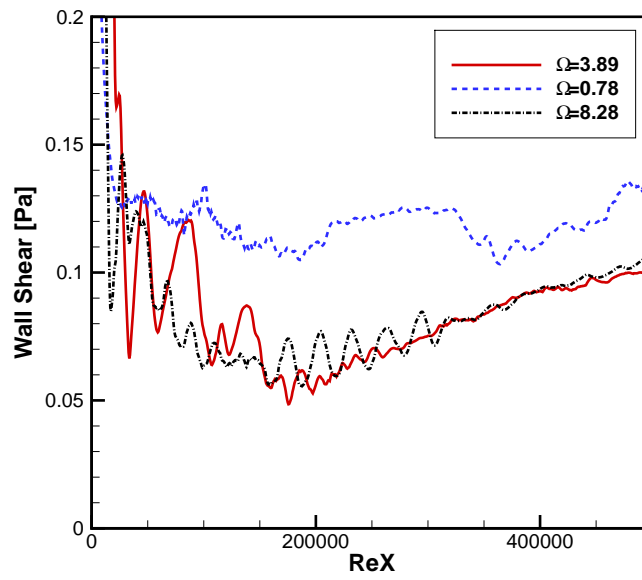


Figure 10.17: Temporally averaged wall shear.

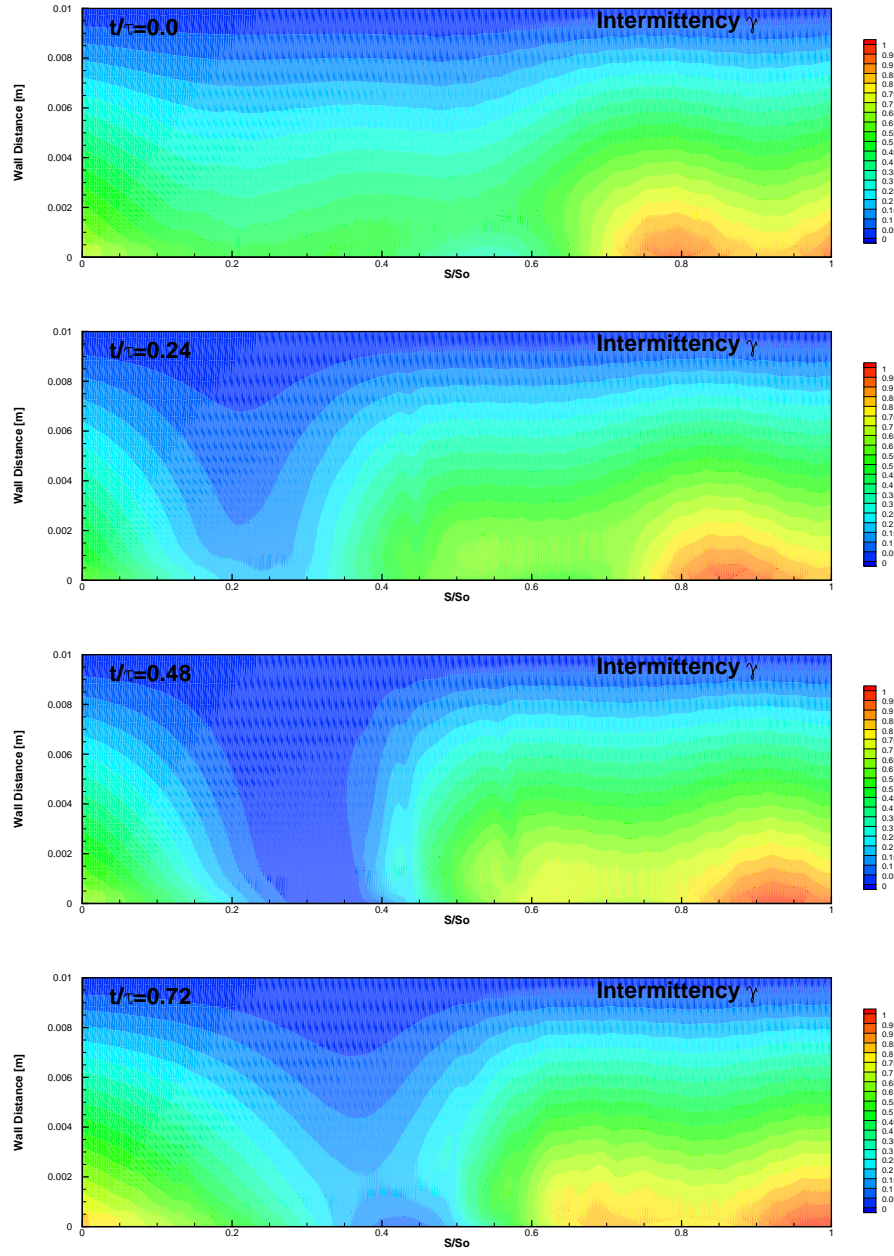


Figure 10.18: Contours of intermittency at different time phases for $\Omega = 0.77$.

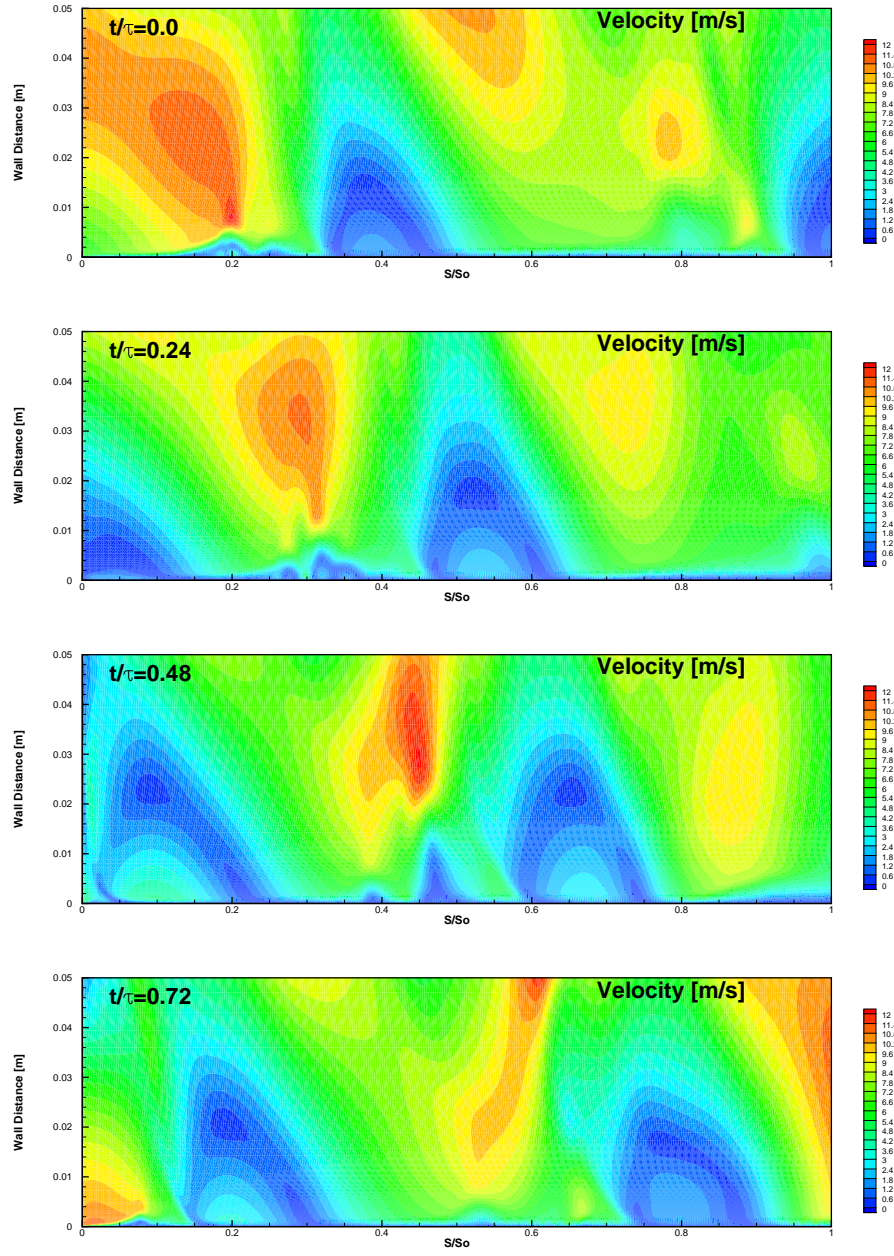


Figure 10.19: Contours of absolute velocity at different time phases for $\Omega = 0.77$.

10.6 Blade Cascade Results

In addition to the extensive curved plate testing and simulation, a blade cascade was also examined by Chakka [114], the geometry used in this cascade is shown in figure 10.20 below. The mesh uses 1.2×10^6 nodes and a 1:1 periodic matching of the top and bottom periodicity patches.

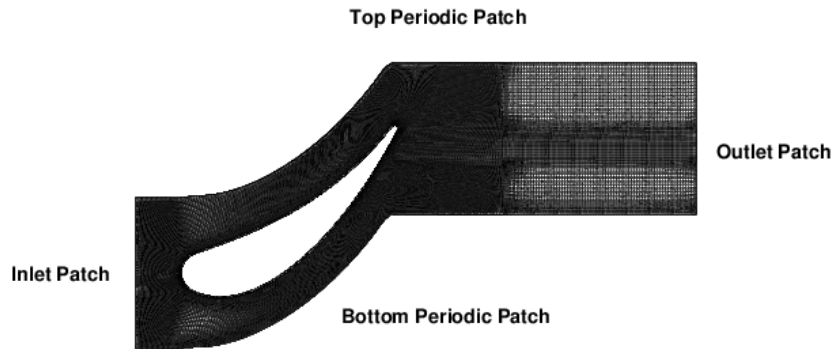


Figure 10.20: Numerical mesh used for blade cascade simulations.

Although a exact model formulation in terms of transition start and end axial Reynolds numbers are not listed in [114] a steady state transition model similar to the formulation in the previous section and based on the Chakka and Schobeiri [53] model was constructed via fortran routines and employed into the commercial solver CFX. The transition start and end points were based on Reynolds numbers obtained from the curved plate experiment for the steady case. The obtained results were compared to the Menter $\gamma - Re_\theta$ model. The boundary conditions used in this simulation are listed in table 10.5

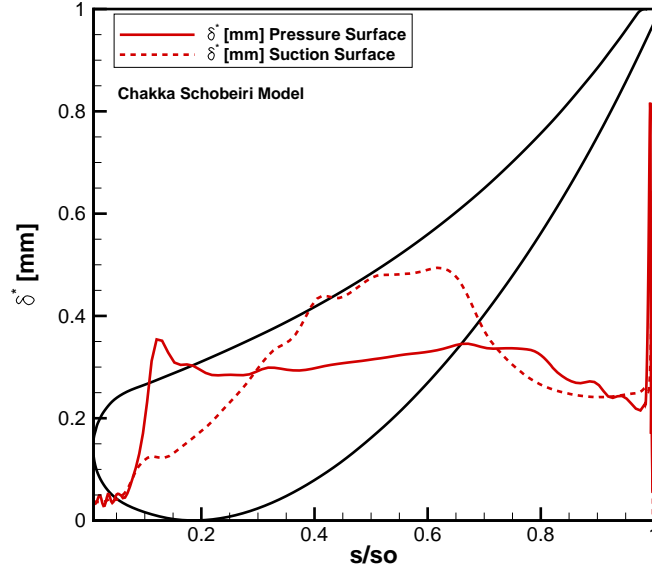
10.6.1 Blade Cascade: Chakka and Schobeiri Model

As stated the model used for the curved plate was again employed here, the case was run in steady mode, and the resultant data analyzed.

Table 10.5: Blade cascade boundary condition for RANS simulation.

Case	Inlet Velocity [m/s]	Exit Pressure [Pa]	Turbulence Model
1	15.0	Atmospheric 0.0 Pa Gauge	Chakka and Schobeiri [53]
2	15.0	Atmospheric 0.0 Pa Gauge	$\gamma - Re_\theta$ [80]

Figure 10.21 shows the displacement thickness for both pressure and suction surfaces, with a clear suction surface transition point identified at an S/S_o value of just under 0.6. The pressure surface on the other hand exhibits transition at a much later point, this does not clearly appear in the displacement thickness plot, where a subtle reduction in displacement thickness occurs at $S/S_o = 0.6$. This is observed clearly when plotting the heat transfer coefficient, as shown in figure 10.25. The intermittency function applied onto the blade is shown in, figure 10.23, where transition, as expected occurs slightly earlier for the suction surface rather than the pressure surface.

**Figure 10.21:** Displacement thickness δ^* at the pressure and suction surfaces, as predicted by the Chakka and Schobeiri model.

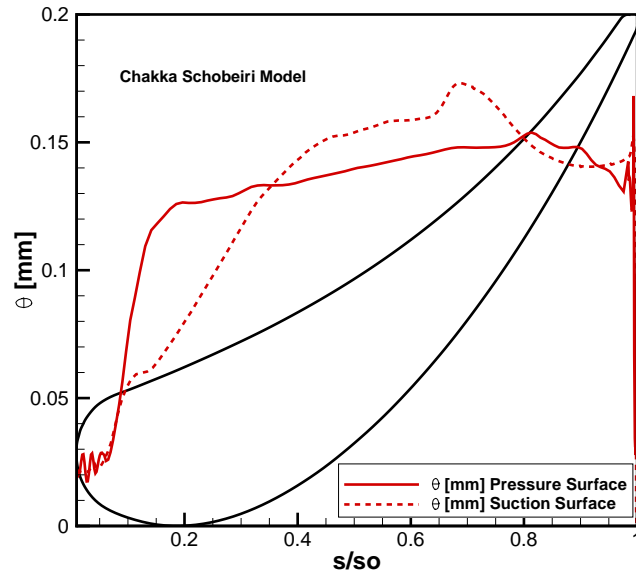


Figure 10.22: Momentum thickness θ at the pressure and suction surfaces as predicted by the Chakka and Schobeiri model.

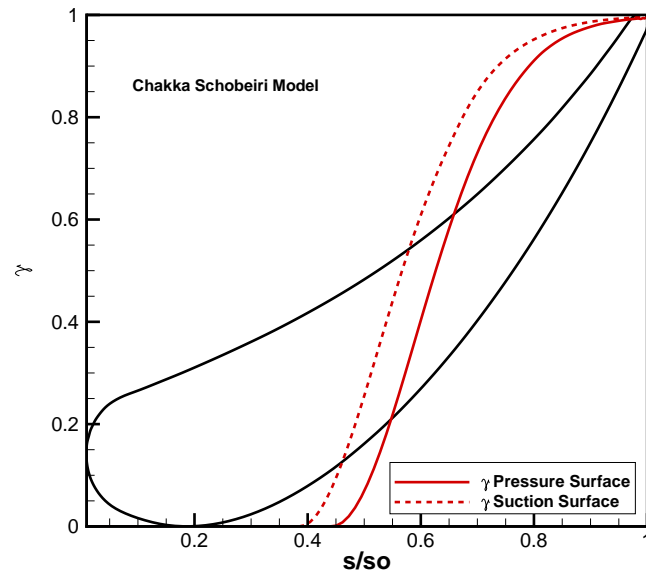


Figure 10.23: Intermittency function γ applied at the pressure and suction surfaces using the Chakka and Schobeiri model.

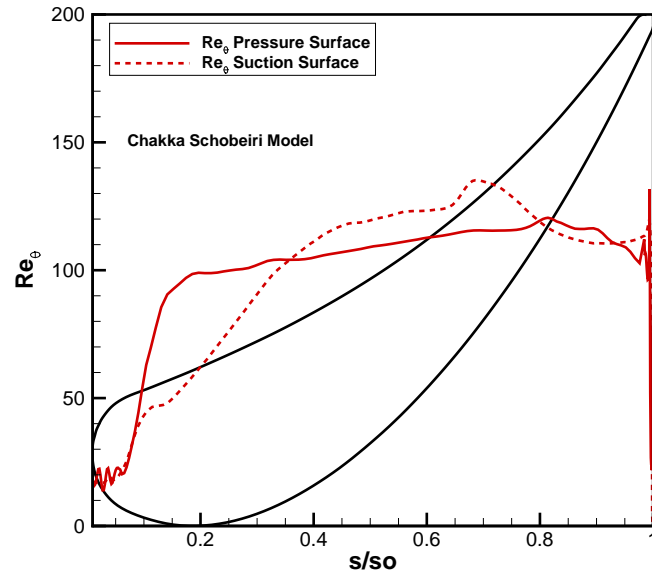


Figure 10.24: Momentum thickness Reynolds number Re_θ at the pressure and suction surfaces as predicted by the Chakka and Schobeiri model.

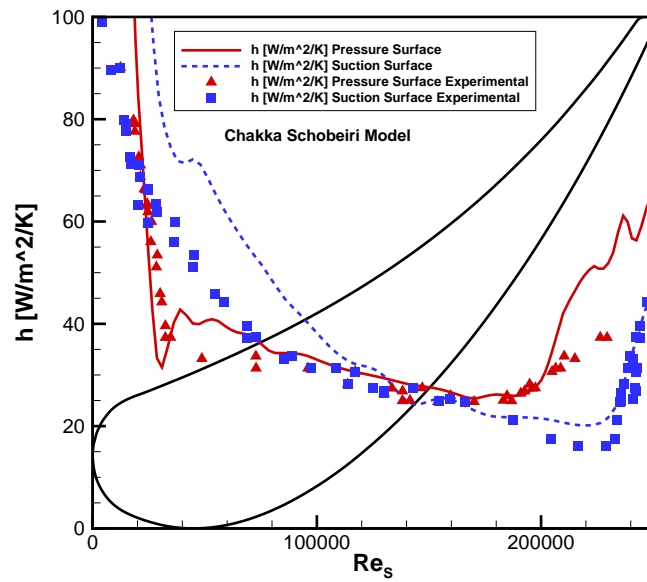


Figure 10.25: Heat transfer coefficient h at the pressure and suction surfaces as predicted by the Chakka and Schobeiri model.

The same pattern observed with the steady plate transition simulations, is again observed here, the effect of transition is abrupt and occurs right when the intermittency function γ reaches the value 1, or a fully turbulent flow regime. This is due to the inclusion of the intermittency function into the k equation, where it is a multiplier to the production term, rather than a multiplier on the turbulent viscosity term μ_t . The transition in the steady case using the Chakka and Schobeiri model is consistent with the experimental results presented in [114]. Figure 10.24 which shows the momentum thickness Reynolds number Re_θ as a function of normalized axial coordinates, shows that transition starts at an approximate value of 80 for the momentum thickness Reynolds number, and not the approximate value of 200 in the case of the plate. This is an interesting point since the author has observed that in most instances the $\gamma - Re_\theta$ model requires a value of at least Re_θ before transition is triggered. The correlation for the critical momentum thickness Reynolds number is listed in [109], and is a strongly dependent on inlet turbulence intensity as well as other various factors, however for the 'steady' inlet turbulence values used for both the plate and the blade cascade, the critical momentum thickness number is 200 for the $\gamma - Re_\theta$ model. This explains why in the next section, transition is not triggered when the model explained in [109] is employed.

10.6.2 Blade Cascade: $\gamma - Re_\theta$ Model

As explained above, for the inlet turbulence intensity used in both the cascade and plate simulations, which in essence model the experimental inlet turbulent intensity, the $\gamma - Re_\theta$ model is not triggered and the flow characteristics appear to be fully turbulent. This is clearly shown in the displacement thickness plot, figure 10.26, the momentum thickness plot figure 10.27 and finally the heat transfer coefficient figure 10.30. The resultant intermittency on the blade walls is 0.02, in figure 10.28, which really suggests a fully turbulent flow, since the intermittency function in this model is subjected to a zero gradient boundary condition at all walls [109].

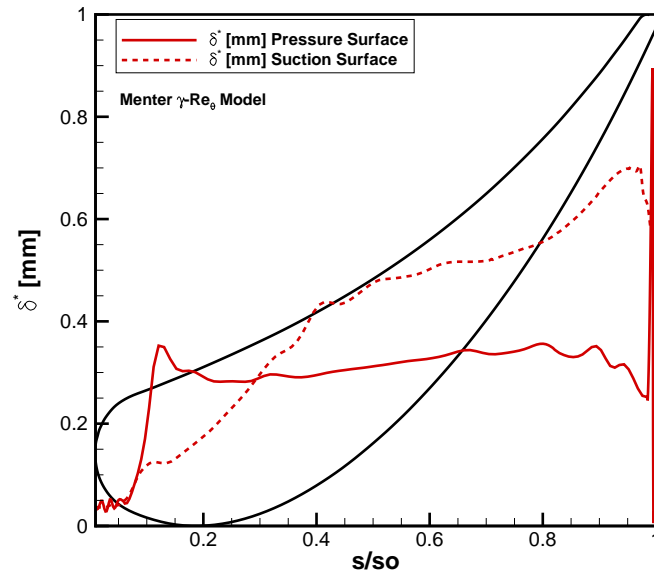


Figure 10.26: Displacement thickness δ^* at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.

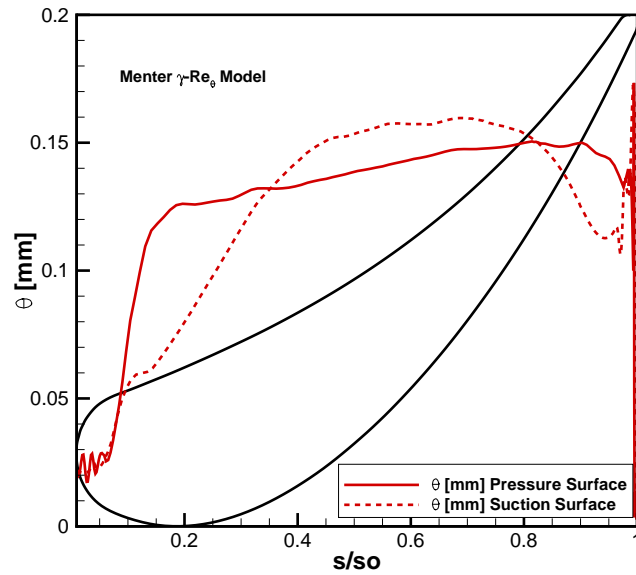


Figure 10.27: Momentum thickness θ at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.

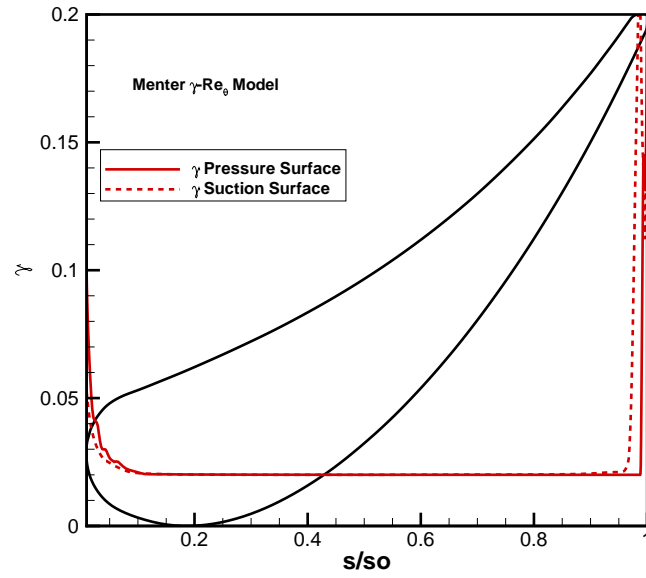


Figure 10.28: Intermittency function γ applied at the pressure and suction surfaces using the $\gamma - Re_\theta$ model.

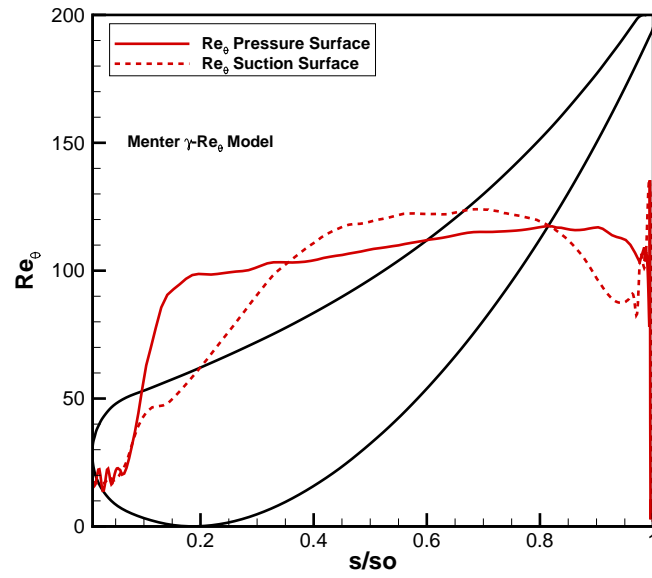


Figure 10.29: Momentum thickness Reynolds number Re_θ at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.

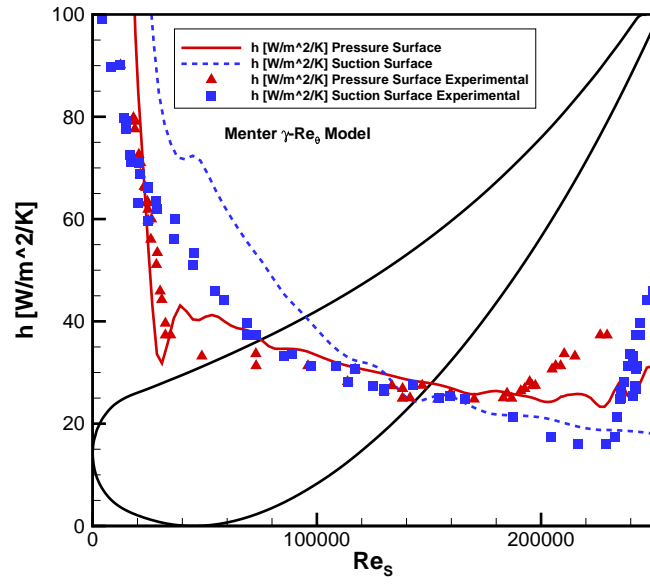


Figure 10.30: Heat transfer coefficient h at the pressure and suction surfaces as predicted by the $\gamma - Re_\theta$ model.

11. DISCUSSION:TRANSITION MODELING

The transition model of Chakka and Schobeiri [53] was implemented into the finite volume based numerical solver CFX. This model was used to simulate the natural transition as well as the wake induced transition of flow over a curved plate, moreover the model in its steady state mode was used to model the natural transition over a blade cascade as was done by Chakka [114]. The implementation leveraged the intermittency functions available in CFX that are used as multipliers to the production term in the k equation of the SST model by Menter [80]. Overall the results, especially in the plate case, showed that in steady state mode, the $\gamma - Re_\theta$ model clearly predicts a much earlier transition point than the Chakka and Schobeiri model. Moreover, it was observed that the transition is triggered at a momentum thickness Reynolds number of approximately 200 for the plate case. When this was used as a fixed value to trigger transition in the $\gamma - Re_\theta$ model location of transition was further upstream than the regular $\gamma - Re_\theta$ and the Chakka and Schobeiri model.

The Chakka and Schobeiri model was also run in unsteady mode, to model various reduced frequency of upstream wakes impinging onto the plate surface, in this case the relative intermittency Γ concept was used as described in [53]. Higher reduced frequency values result in a much earlier transition point, when compared to lower values of reduced frequency. This result correctly image the patterns observed in both Chakka and Schobeiri [53] and Chakka [114]. The model by Chakka and Schobeiri was shown to work extremely well, in both steady state mode and in wake induced transition mode where it provides an extra dimension for which the use of the $\gamma - Re_\theta$ model was not intended. However, in order to better fit within a CFD solver, the resultant intermittency needs to become a turbulent viscosity μ_t multiplier rather than a multiplier on the k equation production term as shown in (10.37). The implementation of the relative intermittency provides that extra capability to incorporated unsteady effects on how they influence the transition to turbulent flow,

however the model is better served by being based on a local value such as the momentum thickness Reynolds number Re_θ rather than the axial Reynolds number as used in [53]. This will facilitate the model's use in various CFD solvers, but will need a transport equation to model the momentum thickness Reynolds number. The model will need further experimental or DNS based calibration with various geometries and operational parameters such as turbulence intensity, reduced frequency, flow incidence to name a few. However the formulation of this model provides an excellent foundation to build upon to provide a accurate description of laminar to turbulent transition under natural conditions as well as wake induced conditions that are part and parcel of turbomachinery flows, and allowing designers are more accurate capability to design their blade shapes for unsteady effects which RANS based CFD is unable to provide.

12. CONCLUSIONS

Turbomachinery flows are extremely complex, for their highly three-dimensional and unsteady nature, CFD has advanced at a fast rate, and has now become a de facto tool in the turbomachinery design process. However, can CFD in its current state of the art be used as a predictor of a specific design's performance? This is what this study attempted to answer, and the conclusion is that CFD cannot be used as a pure predictor of performance in the turbomachinery arena. Rather it is best to use CFD to compare on a relative basis, the performance of one design to another. Such a scenario could be looking at the penalty incurred in efficiency points when a design is switched from using shrouded stators and rotors to using tip clearances.

Three different blade designs were studied, both experimentally and numerically, they varied from being a two stage design to a three stage design using in one configuration, cylindrical blades and in another a positive compound lean in all blade rows. CFD simulations for all designs in both steady and unsteady mode were used. The cases were run at both design operating conditions and off-design conditions which create high blade incidence and the high potential for separated flow, the CFD was in most cases within 2 points of efficiency when compared to experimental data at design conditions and within 3 points at off-design conditions. The general trend was the CFD performance predictions are highly optimistic, which is expected, mainly because, CFD especially in RANS mode does not capture the same wake structure at the trailing edge of each blade row, when compared to the experimental data. Smaller wakes which are shown to dissipate faster axially result in a higher efficiency prediction when compared to experimental data which show larger wake patterns or higher loss. The reasons for the the various deficiencies are discussed in detail in this dissertation as well as in Schobeiri et al. [2], the main factors can be attributed to:

- Modeling or numerical errors.

- The use of mixing planes to facilitate RANS CFD to model turbomachinery flow which is inherently unsteady in nature.
- Turbulence modeling is used to simplify the complex and chaotic nature of turbulence. In order to avoid turbulence modeling errors, direct numerical simulations (DNS) can be used, however they require a grid at the Kolmogorov length scale, consequently immense computational resources which are not practical for turbomachinery flows due to their high Reynolds numbers. A compromise with Large Eddy Simulation (LES) can be used, however, the use of LES to model a multistage turbine or compressor is not fully feasible yet from a computational point of view for design purposes. The emerging use of graphical processing units (GPU) based parallelization may provide the capability to model turbomachinery with higher order turbulence modeling in the near future.
- Transition modeling, used to model the laminar-to-turbulent physics that occurs quite frequently in turbomachinery, especially in low pressure turbines.
- When comparing to experimental data and using such data as boundary conditions, at times certain values such as inlet turbulence intensity may not be measured and therefore an accurate indication of such parameters may not be clear. Inlet turbulent intensity has a large effect on turbulence modeling as well as transition modeling specifically where it will directly influence the location of transition predicted by the code.

Thus, there are many things to consider when using CFD to model turbomachinery flows, although it has been observed that CFD can provide great insight when used to compare between one design to another, and the relative change in performance or flow physics from one design point to another, or from one blade design or feature to another. CFD is still a crucial tool in the turbomachinery design process as well as for research purposes, its advantages and potential pitfalls must be fully understood by users in the turbomachinery arena, particularly young designers.

Another focus of this study was to look at transition modeling, and specifically wake induced transition which, is an important feature of turbomachinery flow physics. The upstream rotor blade row will impart an a set of unsteady wakes that impinge upon the downstream stator row and will certainly influence the general turbulence characteristics of the flow as well as the transition from laminar to turbulent flow. A model was proposed by Chakka and Schobeiri [53] to encompass wake induced effects into transition modeling. The model was implemented into a CFD solver and run for various cases both steady and unsteady showing good correspondence with the experimental results of Chakka[114] and Chakka and Schobeiri [53]. Potential minor improvements to the model are suggested including using a momentum thickness Reynolds number basis to describe the model versus the current axial Reynolds number formulation.

REFERENCES

- [1] GE-Aviation, 2013. GEnx-2B Turbofan. <http://www.geaviation.com/engines/commercial/genx/genx-2b.html>.
- [2] Schobeiri, M., Abdelfattah, S., and Chibli, H., 2012. “Investigating the Cause of Computational Fluid Dynamics Deficiencies in Accurately Predicting the Efficiency and Performance of High Pressure Turbines: A Combined Experimental and Numerical Study”. *Journal of Fluids Engineering*, **134** (10).
- [3] Schobeiri, M., and Abdelfattah, S., 2013. “On the Reliability of RANS and URANS Numerical Results for HP-Turbine Simulations: A Benchmark Experimental and Numerical Study on Performance and Interstage Flow Behavior of High Pressure Turbines at Design and Off-Design Conditions Using Two Different Turbine Designs”. *Accepted for Publication in the ASME Journal of Turbomachinery*.
- [4] Denton, J. D., 2010. “Some Limitations of Turbomachinery CFD”. GT2010-22540, presented at ASME TURBO EXPO 2010, Glasgow, UK, June 14-18, 2010.
- [5] Gerolymos, G., and Hanisch, C., 1999. “Multistage three dimensional Navier-Stokes computation of off-design operation of a four stage turbine”. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Mechanical Engineering Science*, **213**(2), pp. 243–261.
- [6] Gerolymos, G., Neubauer, J., Sharma, V., and Vallet, I., 2002. “Improved Prediction of Turbomachinery Flows Using Near-Wall Reynolds-Stress Model”. *Journal of Turbomachinery*, **124**, pp. 86–99.
- [7] Haldeman, C., Dunn, M., Abhari, R., Johnson, P., and Montesdeoca, X., 2000. “Experimental and Computational Investigation of the Time-Averaged and Time-Resolved Pressure Loading on a Vaneless Counter-Rotating Turbine”. 2000GT-0445, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
- [8] Schobeiri, M. T., 2012. *Turbomachinery Flow Physics and Dynamic Performance*.

Springer-Verlag, Berlin, Germany.

- [9] Denton, J. D., 1993. “The 1993 IGTI Scholar Lecture: Loss Mechanisms in Turbo-machines”. *Journal of Turbomachinery*, **115**(4), pp. 621–656.
- [10] Sharma, K., 2011. “Inter-Stage and Performance Test of a Two-Stage High Pressure Turbine”. Masters thesis, Texas A&M University.
- [11] Schlichting, H., 1978. *Boundary Layer Theory*. McGraw-Hill, New York.
- [12] Porreca, L., Behr, T., Schlienger, J., Kalfas, A. I., Abhari, R. S., Ehrhard, J., and Janke, E., 2005. “Fluid Dynamics and Performance of Partially and Fully Shrouded Axial Turbines”. *Journal of Turbomachinery*, **127**(4), pp. 668–678.
- [13] Adami, P., Milli, A., Martelli, F., and Cecchi, S., 2006. “Comparison of Different Shroud Configurations in High-Pressure Turbines Using Unsteady CFD”. GT2006-90442, presented at ASME TURBO EXPO 2006, Barcelona, Spain, May 8-11, 2006.
- [14] Langston, L., 2001. “Secondary Flows in Axial Turbines—A Review”. *Annals of the New York Academy of Sciences*, **934**(1), pp. 11–26.
- [15] Sharma, O. P., and Butler, T. L., 1987. “Predictions of Endwall Losses and Secondary Flows in Axial Flow Turbine Cascades”. *Journal of Turbomachinery*, **109**(2), pp. 229–236.
- [16] Langston, L., Nice, M., and Hooper, R., 1977. “Three-Dimensional Flow Within a Turbine Cascade Passage”. *Journal of Engineering for Power*, **99**, pp. 21–28.
- [17] Sieverding, C., and Van den Bosche, 1983. “The Use of Colored Smoke to Visualize Secondary Flows in a Turbine-Blade Cascade”. *Journal of Fluid Mechanics*, **134**, pp. 85–89.
- [18] Wang, H., Olson, S., Goldstein, R., and Eckert, E., 1997. “Flow Visualization in a Linear Turbine Cascade of High Performance Turbine Blades”. *Journal of Turbomachinery*, **119**, pp. 1–8.
- [19] Baker, C., 1979. “The Laminar Horseshoe Vortex”. *Journal of Fluid Mechanics*, **95**, pp. 347–367.
- [20] Eckerle, W. A., and Langston, L. S., 1987. “Horseshoe Vortex Formation Around a

- Cylinder”. *Journal of Turbomachinery*, **109**(2), pp. 278–285.
- [21] Eckerle, W. A., and Awad, J., 1991. “Effect of Freestream Velocity on the Three-Dimensional Separated Flow Region in Front of a Cylinder”. *Journal of Fluids Engineering*, **113**, pp. 37–44.
 - [22] Hah, C., 1984. “A Navier-Stokes Analysis of Three-Dimensional Turbulent Flows Inside Turbine Blade Rows at Design and Off-Design Conditions”. *Journal of Engineering for Gas Turbines and Power*, **106**(2), pp. 421–429.
 - [23] Dorney, D. J., and Davis, R. L., 1992. “Navier–Stokes Analysis of Turbine Blade Heat Transfer and Performance”. *Journal of Turbomachinery*, **114**(4), pp. 795–806.
 - [24] Hartland, J. C., Gregory-Smith, D. G., Harvey, N. W., and Rose, M. G., 2000. “Nonaxisymmetric Turbine End Wall Design: Part II—Experimental Validation”. *Journal of Turbomachinery*, **122**(2), pp. 286–293.
 - [25] Gregory-Smith, D. G., and Cleak, J. G. E., 1992. “Secondary Flow Measurements in a Turbine Cascade with High Inlet Turbulence”. *Journal of Turbomachinery*, **114**(1), pp. 173–183.
 - [26] Denton, J., 1975. A Time Marching Method for Two-and Three-Dimensional Blade to Blade Flows. Tech. rep., H.M. Stationery Office.
 - [27] Adamczyk, J. J., 2000. “Aerodynamic Analysis of Multistage Turbomachinery Flows in Support of Aerodynamic Design”. *Journal of Turbomachinery*, **122**, pp. 189–217.
 - [28] Hall, E., 1998. “Aerodynamic Modelling of Multistage Compressor Flow Fields Part 2: Modeling Deterministic Stresses”. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, **212**(2), pp. 91–107.
 - [29] Adamczyk, J., Celestina, M., Beach, T., and Barnett, M., 1990. “Simulation of Three-Dimensional Viscous Flow Within a Multistage Turbine”. *Journal of Turbomachinery*, **112**, pp. 370–376.
 - [30] Denton, J., 1992. “The Calculation of Three-Dimensional Viscous Flow Through Multistage Turbomachines”. *Journal of Turbomachinery*, **114**, pp. 18–26.
 - [31] Dawes, W., 1992. “Towards Improved Throughflow Capability: The Use of Three

- Dimensional Viscous Flow Solvers in a Multistage Environment”. *Journal of Turbomachinery*, **114**, pp. 8–17.
- [32] Mulac, R., and Adamczyk, J., 1992. “The Numerical Simulation of a High-Speed Axial Flow Compressor”. *Journal of Turbomachinery*, **114**, pp. 517–527.
 - [33] Adamczyk, J. J., 2000. “Aerodynamic Analysis of Multistage Turbomachinery Flows in Support of Aerodynamic Design”. *Journal of Turbomachinery*, **122**(2), pp. 189–217.
 - [34] Chima, R., 1998. “Calculation of Multistage Turbomachinery Using Steady Characteristic Boundary Conditions”. *AIAA Paper 98-0968*.
 - [35] Kunz, R. F., and Lakshminarayana, B., 1992. “Three-Dimensional Navier–Stokes Computation of Turbomachinery Flows Using an Explicit Numerical Procedure and a Coupled k–epsilon Turbulence model”. *Journal of Turbomachinery*, **114**(3), pp. 627–642.
 - [36] Dawes, W. N., 1992. “The Simulation of Three-Dimensional Viscous Flow in Turbomachinery Geometries Using a Solution-Adaptive Unstructured Mesh Methodology”. *Journal of Turbomachinery*, **114**(3), pp. 528–537.
 - [37] Gerolymos, G. A., and Vallet, I., 2007. “Robust Implicit Multigrid Reynolds-Stress Model Computation of 3D Turbomachinery Flows”. *Journal of Fluids Engineering*, **129**(9), pp. 1212–1227.
 - [38] Chen, T., Vasanthakumar, P., and He, L., 2000. “Analysis of Unsteady Blade Row Interaction using Nonlinear Harmonic Approach”. 2000GT-0431, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
 - [39] Giangiacomo, P., Michelassi, V., and Martelli, F., 2000. “Analysis of the Mixing Plane Interface Between Stator and Rotor of a Transonic Axial Turbine Stage”. 2000GT-0633, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
 - [40] Ingram, G., and Gregory-Smith, D., 2004. “Investigation of a Novel Secondary Flow Feature in a Turbine Cascade with End Wall Profiling”. GT2004-53589, presented

- at ASME TURBO EXPO 2004, Vienna, Austria, June 14-17, 2004.
- [41] Rosic, B., Denton, J. D., and Pullan, G., 2005. “The Importance of Shroud Leakage Modelling in Multistage Turbine Flow Calculations”. GT2005-68459, presented at ASME TURBO EXPO 2005, Reno-Tahoe, Nevada, USA, June 6-9, 2005.
 - [42] Muth, B., Schwarz, M., Niehuis, R., and Franke, M., 2009. “Investigation of CFD Prediction Capabilities for Low Reynolds Turbine Aerodynamics”. GT2009-59306, presented at ASME TURBO EXPO 2009, Orlando, Florida, USA, June 8-12, 2009.
 - [43] Barter, J., Chen, J., and Vitt, P., 2000. “Interaction Effects in a Transonic Turbine Stage”. GT2000-0376, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
 - [44] Van Zante, D., Chen, J., Hathaway, M., and Chriss, R., 2009. “The Influence of Compressor Blade Row Interaction Modeling on Performance Estimates From Time-Accurate, Multistage, Navier-Stokes Simulations”. *Journal of Turbomachinery*, **130**, pp. 011009–1–011009–10.
 - [45] Pullan, G., 2006. “Secondary Flows and Loss Caused by Blade Row Interaction in a Turbine Stage”. *Journal of Turbomachinery*, **128**(3), pp. 484–491.
 - [46] Casciaro, C., Treiber, M., and Sell, M., 2000. “Unsteady Transport Mechanisms in an Axial Turbine”. *Journal of Turbomachinery*, **122**(4), pp. 604–612.
 - [47] Meyer, R., 1958. “The Effect of Wakes on the Transient Pressure and Velocity Distributions in Turbomachines”. *Journal of Basic Engineering*, **80**, pp. 1544–1552.
 - [48] Rose, M. G., and Harvey, N. W., 2000. “Turbomachinery Wakes: Differential Work and Mixing Losses”. *Journal of Turbomachinery*, **122**(1), pp. 68–77.
 - [49] Denos, R., Arts, T., Paniagua, G., Michelassi, V., and Martelli, F., 2000. “Investigation of the Unsteady Rotor Aerodynamics in a Transonic Turbine Stage”. 2000GT-0435, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
 - [50] Ghannam, B. A., and Shaw, R., 1981. “Natural Transition of Boundary Layers — The Effects of Turbulence, Pressure Gradient and Flow History”. *Journal of Mechanical Engineering Science*, **22**(5), pp. 213–228.

- [51] Michelassi, V., Martelli, F., Denos, R., and Sieverding, C., 1999. “Unsteady heat transfer in stator-rotor interaction by two equation turbulence model”. *Journal of Turbomachinery*, **121**, pp. 436–447.
- [52] Halstead, D., Wisler, D., Okiishi, T., Walker, G., Hodson, H., and Shin, H., 1997. “Boundary Layer Development in Axial Compressors and Turbines Part 3 of 4: LP Turbines”. *Journal of Turbomachinery*, **119**, pp. 225–237.
- [53] Chakka, P., and Schobeiri, M. T., 1999. “Modeling Unsteady Boundary Layer Transition on a Curved Plate Under Periodic Unsteady Flow Conditions: Aerodynamic and Heat Transfer Investigations”. *Journal of Turbomachinery*, **121**(1), pp. 88–97.
- [54] Clark, J., Stetson, G., Magge, S., Ni, R., Haldeman, C., and Dunn, M., 2000. “The Effect of Airfoil Scaling on the Predicted Unsteady Loading on the Blade of a 1 and 1/2 Stage Transonic Turbine and a Comparison with Experimental Results”. 2000GT-0446, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
- [55] Bohn, D. E., Ren, J., and Tummers, C., 2005. “Unsteady 3D-Numerical Investigation of the Influence of the Blading Design on the Stator-Rotor Interaction in a 2-Stage Turbine”. GT2005-68115, presented at ASME TURBO EXPO 2005, Reno-Tahoe, Nevada, USA, June 6-9, 2005.
- [56] Gehrler, A., Lang, H., Mayrhofer, N., and Woisetschlager, J., 2000. “Numerical and Experimental Investigation of Trailing Edge Vortex Shedding Downstream of a Linear Turbine Cascade”. 2000GT-0434, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
- [57] Vand, M. H., and Songtao, W., 2005. “Numerical Study of the Effects of Bowed Blades on Aerodynamic Characteristics in a High Pressure Turbine”. GT2005-68214, presented at ASME TURBO EXPO 2005, Reno-Tahoe, Nevada, USA, June 6-9, 2005.
- [58] Yoon, S., Curtis, E., Denton, J., and Longley, J., 2010. “The Effect of Clearance on Shrouded and Unshrouded Turbines at Two levels of Reaction”. GT2010-22541,

- presented at ASME TURBO EXPO 2010, Glasgow, UK, June 14-18, 2010.
- [59] El-Ghandour, M., Shatat, M., and Nakamura, Y., 2012. “On the Physics of Vortex Formation at the Tip of a Turbine Blade”. GT2012-70128, presented at ASME TURBO EXPO 2012, Copenhagen, Denmark, June 11-15, 2012.
 - [60] Menter, F., 1993. “Zonal Two Equation $k-\omega$ Turbulence Models for Aerodynamic Flows”. *AIAA Paper 93-2906*.
 - [61] Menter, F., 1994. “Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications”. *AIAA Journal*, **32**, pp. 1598–1605.
 - [62] Mumic, F., Ljungkruna, L., and Sunden, B., 1992. “Numerical Simulations of Heat Transfer and Fluid Flow for a Rotating High-Pressure Turbine”. AIAA-92-0393, presented at 30th Aerospace Sciences Meeting and Exhibit, Reno, NV, January 6-9, 1992.
 - [63] Sanz, W., Pieringer, P., Baumgartner, W., Edelbauer, W., Pilz, C., Neumayer, F., and Rossi, E., 2004. “CFD Simulation of the Losses in an Inlet Guide Vane of a Transonic Test Turbine”. GT2004-53846, presented at ASME TURBO EXPO 2004, Vienna, Austria, June 14-17, 2004.
 - [64] Mischo, B., Behr, T., and Abhari, R. S., 2006. “Flow Physics and Profiling of Recessed Blade Tips: Impact on Performance and Heat Load”. GT2006-91074, presented at ASME TURBO EXPO 2006, Barcelona, Spain, May 8-11, 2006.
 - [65] Adami, P., Milli, A., Martelli, F., and Cecchi, S., 2006. “Comparison of Different Shroud Configurations in High-Pressure Turbines Using Unsteady CFD”. GT2006-90442, presented at ASME TURBO EXPO 2006, Barcelona, Spain, May 8-11, 2006.
 - [66] Medic, G., You, D., and Kalitzin, G., 2012. “On Coupling of RANS and LES for Integrated Computations of Jet Engines”. GT2007-27096, presented at ASME TURBO EXPO 2007, Montreal, Canada, May 14-17, 2007.
 - [67] Reynolds, O., 1883. “On the Experimental Investigation of the Circumstance Which Determine Whether the Motion of Water Shall be Direct or Sinous, and the Law of Resistance in Parallel Channels”. *Phil. Trans. Roy. Soc. A.*, **174**, pp. 935–982.

- [68] Burgers, J., 1924. “The Motion of a Fluid in the Boundary Layer Along a Smooth Surface”. *Proc. First International Congress for Applied Mechanics*, **174**, pp. 935–982.
- [69] van der Hegge Zijnen, B., 1924. “Measurements of the Velocity Distribution in the Boundary Layer Along a Plane Surface”. Masters thesis, Delft University.
- [70] Rayleigh, L., 1913. “On the Stability of Certain Fluid Motions”. *Proc. Math. Soc. London*, **197**.
- [71] Prandtl, L., 1921. “Über die Entstehung der Turbulenz”. *ZAMM*, **I**, pp. 431–436.
- [72] Schlichting, H., 1933. “Zur Entstehung der Turbulenz bei der Plattenströmung. Nachr. Ges. Wiss. Göttingen”. *ZAMM*, **13**, pp. 171–174.
- [73] Tollmien, W., 1931. Über die Entstehung der Turbulenz 1. Mitt. Nachr. Ges. Wiss. Göttingen. Tech. rep., NACA TM-609.
- [74] Mayle, R. E., 1991. “The 1991 IGTI Scholar Lecture: The Role of Laminar-Turbulent Transition in Gas Turbine Engines”. *Journal of Turbomachinery*, **113**(4), pp. 509–536.
- [75] Praisner, T. J., and Clark, J. P., 2007. “Predicting Transition in Turbomachinery—Part I: A Review and New Model Development”. *Journal of Turbomachinery*, **129**(1), pp. 1–13.
- [76] Emmons, H., 1951. “The Laminar-Turbulent Transition in Boundary Layers-Part I”. *J. Aero. Sci.*, **18**, pp. 490–498.
- [77] Dhawan, S., and Narashima, R., 1988. “Some Properties of Boundary Layer Flow During the Transition from Laminar to Turbulent Motion”. *Journal of Fluid Mechanics*, **3**, pp. 418–436.
- [78] Morkovin, M., 1969. “On the Many Faces of Transition”. *Viscous Drag Reduction*, C.S. Wells, Ed. Plenum Press, New York, pp. 1–31.
- [79] Gostelow, J. P., and Blunden, A. R., 1989. “Investigations of Boundary Layer Transition in an Adverse Pressure Gradient”. *Journal of Turbomachinery*, **111**(4), pp. 366–374.

- [80] Menter, F. R., Langtry, R. B., Likki, S. R., Suzen, Y. B., Huang, P. G., and Völker, S., 2006. “A Correlation-Based Transition Model Using Local Variables—Part I: Model Formulation”. *Journal of Turbomachinery*, **128**(3), pp. 413–422.
- [81] Langtry, R. B., Menter, F. R., Likki, S. R., Suzen, Y. B., Huang, P. G., and Völker, S., 2006. “A Correlation-Based Transition Model Using Local Variables—Part II: Test Cases and Industrial Applications”. *Journal of Turbomachinery*, **128**(3), pp. 423–434.
- [82] Gier, J., Ardey, S., and Heisler, A., 2000. “Analysis of Complex Three-Dimensional Flow in a Three-Stage LP Turbine by Means of Transitional Navier-Stokes Simulation”. 2000GT-0645, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
- [83] Hu, J., and Fransson, T. H., 2000. “Numerical Performance of Transition Models in Different Turbomachinery Flow Conditions: A Comparative Study”. 2000GT-0520, presented at ASME TURBO EXPO 2000, Munich, Germany, May 8-11, 2000.
- [84] Babajee, J., and Arts, T., 2012. “Investigation of the Laminar Separation-Induced Transition with the $\gamma\text{-}\widetilde{Re}_{\theta t}$ Transition Model on Low-Pressure Turbine Rotor Blades at Steady Conditions”. GT2012-70128, presented at ASME TURBO EXPO 2012, Copenhagen, Denmark, June 11-15, 2012.
- [85] Schobeiri, M. T., 2010. *Fluid Mechanics for Engineers: A Graduate Textbook*. Springer-Verlag, Berlin, Germany.
- [86] Cebeci, T., and Smith, A., 1974. *Analysis of Turbulent Boundary Layers*. Academic Press, New York.
- [87] Baldwin, B., and Lomax, H., 1978. Thin Layer Approximation and Algebraic Model for Separated Turbulent flows. *AIAA Paper* 78-0257.
- [88] Jones, W., and Launder, B., 1973. “The Calculation of Low-Reynolds Number Phenomena with a Two-Equation Model of Turbulence”. *International Journal of Heat and Mass Transfer*, **16**, pp. 1119–1130.
- [89] Wilcox, D., 1992. Turbulence Modeling for CFD. Tech. rep., DCW Industries, La Canada, CA.

- [90] Wilcox, D., 1993. Turbulence and Transition Modeling for High-Speed Flows. Tech. Rep. CR 191473, NASA.
- [91] Schobeiri, M., Gilarranz, J., and Johansen, E., 2004. “Aerodynamic and Performance Behavior of a Three-Stage High Efficiency Turbine at Design and Off-Design Operating Points”. *International Journal of Rotating Machinery*, **10**, pp. 33–44.
- [92] Schobeiri, M., Suryanarayanan, A., Jermann, C., and Neuenschwander, T., 2004. “A Comparative Aerodynamic and Performance Study of Three-Stage High-Pressure Turbine with 3-D Bowed Blade and Cylindrical Blades”. GT2004-53650, presented at ASME TURBO EXPO 2004, Vienna, Austria, June 14-17, 2004.
- [93] Suryanarayanan, A., Mhetras, S., Schobeiri, M., and Han, J., 2009. “Film-Cooling Effectiveness on a Rotating Blade Platform”. *Journal of Turbomachinery*, **131**, pp. 0110141–1–0110141–12.
- [94] Borello, D., Hanjalic, K., and Rispoli, F., 2007. “Computation of tip-leakage flow in a linear compressor cascade with a second-moment turbulence closure”. *International Journal Heat and Fluid Flow*, **28**, pp. 587–601.
- [95] Dambach, R., Hodson, H., and Huntsman, I., 1998. “An Experimental Study of Tip Clearance Flow in a Radial Inflow Turbine”. 98-GT-467, Presented at International Gas Turbine and Aeroengine Congress and Exhibition 1998, Stockholm, Sweden, June 2-5, 1998.
- [96] Soljander, S., and Armud, K., 1987. “Effects of Tip Clearance on Blade Loading in a Planar Cascade of Turbine Blades”. *Journal of Turbomachinery*, **109**, pp. 237–244.
- [97] ANSYS INC., 2009. *ANSYS-CFX Release Documentation*, 12.0 ed.
- [98] Matsunuma, T., Abe, H., and Y.Tsutsui, 1999. “Influence of Turbine Intensity on Annular Turbine Stator Aerodynamics”. *ASME Paper No. 99-GT-151*.
- [99] Denton, J., and Xu, L., 1998. “The Exploitation of Three-Dimensional Flow in Turbomachinery Design”. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, **213**, pp. 125–137.
- [100] Smith, A., and Gamberoni, N., 1956. Transition Pressure Gradient and Stability

- Theory. Tech. Rep. ES26388, Douglas Aircraft Co., El Segundo, California.
- [101] Van Ingen, J., 1956. A Suggested Semi-empirical Method for the Calculation of the Boundary Layer Transition Region. Tech. Rep. UTH-74, Delft University of Technology, Dept. of Aero Eng.
 - [102] Dhawan, S., and Narasimha, R., 1958. “Some Properties of Boundary Layer Flow During Transition from Laminar to Turbulent Motion”. *Journal of Fluid Mechanics*, **3**(4), pp. 418–436.
 - [103] Dey, J., and Narasimha, R., 1992. “An Integral Method for the Calculation of 2D Transitional Boundary Layers”. *Journal of Aircraft*, **27**(10), pp. 859–865.
 - [104] Steelant, J., and Dick, E., 1996. “Modeling of Bypass Transition with Conditioned Navier-Stokes Equations Coupled to an Intermittency Transport Equation”. *International Journal for Numerical Methods in Fluids*, **23**, pp. 193–220.
 - [105] Cho, J., and Chung, M., 1992. “A k-e-v Equation Turbulence Model”. *Journal of Fluid Mechanics*, **237**, pp. 301–322.
 - [106] Huang, P., and Xiong, G., 1998. “Transition and Turbulence Modeling of Low Pressure Turbine Flows”. AIAA-98-0339, Reno, NV.
 - [107] Walters, D., and Leylek, J., 2004. “A New Model for Boundary Layer Transition Using a Single-Point RANS Approach”. *Journal of Turbomachinery*, **126**, pp. 193–202.
 - [108] Mayle, R., and Schulz, A., 1997. “The Path to Predicting Bypass Transition”. *Journal of Turbomachinery*, **119**, pp. 405–411.
 - [109] Langtry, R. B., and Menter, F. R., 2009. “Correlation-based transition modeling for unstructured parallelized computational fluid dynamics codes”. *AIAA Journal*, **47**(12), pp. 2894–2906.
 - [110] Schobeiri, M., and Wright, L., 2003. “Advances in unsteady boundary layer transition research, part i: Theory and modeling”. *International Journal of Rotating Machinery*, **9**(1), pp. 1–9.
 - [111] Schobeiri, M., and Chakka, P., 2002. “Prediction of turbine blade heat transfer and

- aerodynamics using a new unsteady boundary layer transition model”. *International Journal of Heat and Mass Transfer*, **45**(4), pp. 815–829.
- [112] Schobeiri, M., Chakka, P., and Pappu, K., 1998. “Unsteady wake effects on boundary layer transition and heat transfer characteristics of a turbine blade”. In The 1998 International Gas Turbine & Aeroengine Congress & Exhibition.
- [113] Crawford, M., and W.M.Kays, 1976. STAN5(TEXSTAN version) - A Program for Numerical Computation of Two Dimensional Internal and External Boundary Layer Flow. Tech. Rep. CR 2742, NASA.
- [114] Chakka, P., 1998. “Unsteady Wake Effects on Boundary Layer Transition and Heat Transfer Characteristics of Curved Plate and Turbine Blade”. PhD thesis, Texas A&M University, College Station, Texas.

APPENDIX A

TURBINE CFD POSTPROCESS FORTRAN SCRIPTS

A.1 Fortran Program to Analyze RANS Based Turbine CFD Data

```
! TAMU
! TPFL CFD analysis program
! Sherif Abdelfattah (writing a better program for data analysis)
! Copyright 2013 Sherif Abdelfattah <saa2903@gmail.com>
!
module parametr
implicit none
real(8),parameter::pi=3.14159265359,rhub=11.0*25.4e-3,rtip=13.5*25.4e-3
real(8),parameter::r2d=180.0/pi,d2r=pi/180.0
real(8),parameter::ka=1.4,pwr=(ka-1)/ka
integer::norm,ver
save
end module parametr

module turbine
implicit none
integer::nstages,nbr
integer,dimension(50)::nblades
real(8)::rpm,omega,umean,ptin,ptout,psin,psout,htin,htout,co
real(8)::u_co,eta_ts,eta_tt,mexit,ttout,ttin
real(8)::pcor
character(len=4)::rpms

save
end module turbine

module output_dir
character(len=256)::outdir
save
end module output_dir

module pavg
real,dimension(50)::psavg,ptavg,ptravg
end module pavg

program main
use parametr
use turbine
use output_dir

implicit none
character(len=8)::rpms1
character(len=80)::string
real(8)::temp
integer::iq,ns,ns1,ns2,nst !ns is 3 measurement planes,ns1=includes 2 inlet and outlet planes!
integer::n ! n is size of line input
integer::nx !nx is size of contour
integer::i
integer,dimension(50)::nxc
```

```

character(len=3)::cdum

open(unit=3,file='input.deck',status='old')
read(3,*)
read(3,*)rpm
read(3,*)
read(3,*)
read(3,*)
read(3,*)
read(3,*)
read(3,*)
read(3,*)
read(3,*)norm,ver
read(3,*)
read(3,*)cdum
read(3,*)
read(3,*)nstages
read(3,*)
read(3,*)(nblades(i),i=1,nstages*2)
close(3)
write(*,*)'Enter P corrector'
read(*,*)pcor

write(rpms1,'(F8.0)')RPM
rpms1=adjustl(rpms1)
rpms(1:4)=rpms1(1:4)
omega=rpm*2*pi/60
umean=omega*0.5*(rhub+rtip)
nst=3
ns=nstages*2+1
outdir='out_'//rpms
outdir=trim(adjustl(outdir))
nbr=2*nstages

string='mkdir '//trim(adjustl(outdir))
call system(string)

!Get size of line input
open(unit=3,file='Press_station3.dat',status='old')
n=0
do
    read(3,*,iostat=iq)temp
    if(iq /=0)exit
    n = n + 1
end do
close(3)
!Get size of contour input
nx=0
open(unit=3,file='contP_3.dat',status='old')
do
    read(3,*,iostat=iq)temp
    if (iq /= 0)exit
    nx=nx+1
end do
nxc(1)=nx
close(3)

nx=0
open(unit=3,file='contP_4.dat',status='old')
do
    read(3,*,iostat=iq)temp
    if (iq /= 0)exit
    nx=nx+1
end do
nxc(2)=nx
close(3)

nx=0

```



```

open(unit=3,file='contP_5.dat',status='old')
do
    read(3,*,iostat=iq)temp
    if (iq /= 0)exit
    nx=nx+1
end do
nxc(3)=nx
close(3)

nx=maxval(nxc(1:3))

!now call line analysis code
call line_analysis(n,ns,ns1)
call contour_analysis(nx,ns,nst,nxc)

end program

subroutine line_analysis(n,ns,ns1)
use parametr
use turbine
use output_dir
use pavg

implicit none

integer::n,ns,ns1,iq,i,j
character(len=1500)::superstring
character(len=40)::dum_string
character(len=80)::s1,s2,s3,s4,s5,s6,s7,s8
character(len=4)::is,js
character(256)::fout1,fout2,floss,feff,esum,creact
real(8)::vvec_i,num,den,psn,ptrn,ptrn
real(8)::ps3,ps4,ps5,t3,t4,t5,cpa,t4s,t5s,t5s1,dhs1,dhs2,dhs3
real(8)::z_s_avg,z_r_avg,eta_s_avg,eta_r_avg,eta_st_avg,eta_sti_avg
real(8)::pr,lm
real(8),allocatable,dimension(:,:)::span,pt,ps,ptr,mm,mmr
real(8),allocatable,dimension(:,:)::ts,tt,ttr,rho,visc,rc,cp,c
real(8),allocatable,dimension(:,:)::s,hs,hi,ht,htr,ro
real(8),allocatable,dimension(:,:)::u,v,vu,vv,vw,va,vc,vr,vm
real(8),allocatable,dimension(:,:)::w,wu,wv,ww,wa,wc,wr,wm
real(8),allocatable,dimension(:,:)::tke,tet,ted,eddyv
real(8),allocatable,dimension(:,:)::alpha,beta,gamma,gamma1,vb2b,wb2b,wdist,cc,radius

real(8),allocatable,dimension(:)::ss,sr,sst,z_s,z_r
real(8),allocatable,dimension(:)::eta_s,eta_r,eta_st,eta_sti,temp,temp1
real(8),allocatable,dimension(:)::st_flow_coeff,st_load_coeff,react
real(8),dimension(ns)::mdot,psmin,ptmin,ptrmin
real(8),dimension(ns)::psmax,ptmax,ptrmax
real(8),dimension(ns)::vavg,wavg,alphaavg,betaavg,gammaavg
real(8),dimension(ns)::tsavg,ttavg,ttravg,savg,hsavg,htavg,htravg
real(8),dimension(ns)::cpavg,vavg,vaavg,wavg,waavg,vravg,wavg
real(8),dimension(ns)::vcavg,wcavg

dum_string='mkdir '//outdir
dum_string=trim(adjustl(outdir))
call system(outdir)

write(*,*)'allocating memory size: ',n
allocate(span(n,ns),pt(n,ns),ps(n,ns),ptr(n,ns),mm(n,ns),mmr(n,ns),stat=iq)
allocate(ts(n,ns),tt(n,ns),ttr(n,ns),rho(n,ns),visc(n,ns),rc(n,ns),stat=iq)
allocate(cp(n,ns),s(n,ns),hs(n,ns),hi(n,ns),ht(n,ns),htr(n,ns),ro(n,ns),stat=iq)
allocate(u(n,ns),v(n,ns),vu(n,ns),vv(n,ns),vw(n,ns),va(n,ns),vc(n,ns),stat=iq)
allocate(vr(n,ns),vm(n,ns),w(n,ns),wu(n,ns),wv(n,ns),ww(n,ns),wa(n,ns),wc(n,ns),wr(n,ns),wm(n,ns),stat=iq)
allocate(tke(n,ns),ted(n,ns),tfe(n,ns),eddyv(n,ns),radius(n,ns),stat=iq)
allocate(alpha(n,ns),beta(n,ns),gamma(n,ns),gamma1(n,ns),vb2b(n,ns),wb2b(n,ns),stat=iq)
allocate(c(n,ns),wdist(n,ns),cc(n,ns),stat=iq)

```

```

if(iq /=0)then
    write(*,*), 'cannot allocate memory'
    stop
end if

S1='Press_station'
S2='Vel_station'
S3='Turb_station'
S4='ht_station'
do i=1,ns
    write(is,'(I2)')i
    s5=trim(s1)//trim(adjustl(is))//'.dat'
    s6=trim(s2)//trim(adjustl(is))//'.dat'
    s7=trim(s3)//trim(adjustl(is))//'.dat'
    s8=trim(s4)//trim(adjustl(is))//'.dat'
    open(unit=3,file=s5,status='old')
    open(unit=4,file=s6,status='old')
    open(unit=5,file=s7,status='old')
    open(unit=6,file=s8,status='old')
    do j=1,n

        read(3,*)span(j,i),pt(j,i),ptr(j,i),ps(j,i),&
        mm(j,i),mmr(j,i),&
        ts(j,i),tt(j,i),ttr(j,i),rho(j,i),visc(j,i),rc(j,i),cp(j,i),&
        s(j,i),hs(j,i),hi(j,i),ht(j,i),htr(j,i),ro(j,i)

        read(4,*)u(j,i),v(j,i),vu(j,i),vv(j,i),vw(j,i),&
        va(j,i),vc(j,i),vr(j,i),vm(j,i),&
        w(j,i),wu(j,i),wv(j,i),ww(j,i),wa(j,i),wc(j,i),wr(j,i),wm(j,i)

        read(5,*)tke(j,i),ted(j,i),tef(j,i),eddyv(j,i)
        read(6,*)alpha(j,i),beta(j,i),gamma(j,i),vb2b(j,i),wb2b(j,i),&
        wdist(j,i),cc(j,i)

        vb2b(j,i)=sqrt(va(j,i)**2+vc(j,i)**2)
        alpha(j,i)=acos(vc(j,i)/vb2b(j,i))
        gamma(j,i)=atan(vr(j,i)/va(j,i))
        gamma1(j,i)=asin(vr(j,i)/v(j,i))

        radius(j,i)=rhub+span(j,i)*(rtip-rhub)
        u(j,i)=omega*radius(j,i)

        wc(j,i)=vc(j,i)-u(j,i)
        wa(j,i)=va(j,i)
        wr(j,i)=vr(j,i)

        wb2b(j,i)=sqrt(wa(j,i)**2+wc(j,i)**2)
        w(j,i)=sqrt(wb2b(j,i)**2+wr(j,i)**2)
        vvec_i=vb2b(j,i)*cos(alpha(j,i))

        beta(j,i)=acos(wc(j,i)/wb2b(j,i))

        pt(j,i)=ps(j,i)+0.5*rho(j,i)*(v(j,i)*v(j,i))
        ptr(j,i)=ps(j,i)+0.5*rho(j,i)*(w(j,i)*w(j,i)-u(j,i)*u(j,i))

        ht(j,i)=hs(j,i)+0.5*v(j,i)*v(j,i)
        htr(j,i)=hs(j,i)+0.5*(w(j,i)*w(j,i)-u(j,i)*u(j,i))

        c(j,i)=v(j,i)/mm(j,i)
        mmr(j,i)=w(j,i)/c(j,i)

    end do
    close(3)
    close(4)
    close(5)
    close(6)
end do
ps=ps+pcor;pt=pt+pcor;ptr=ptr+pcor

```

```

ps=ps/1000.0;pt=pt/1000.;ptr=ptr/1000.
hs=hs/1000.0;ht=ht/1000.;htr=htr/1000.
s=s/1000.0
allocate(temp(n),temp1(n),stat=iq)
allocate(z_s(n),z_r(n),eta_s(n),eta_r(n),stat=iq)
allocate(eta_st(n),eta_sti(n),ss(n),sr(n),sst(n),stat=iq)
allocate(st_flow_coeff(n,ns),st_load_coeff(n,ns),stat=iq)
allocate(react(n,ns),stat=iq)

do i=1,ns
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*ps(1:n,i)
temp1(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
call integ(n,radius(1:n,i),temp1(1:n),den)
psavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*pt(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
ptavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*ptr(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
ptravg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*s(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
savg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*hs(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
hsavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*ht(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
htavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*htr(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
htravg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*alpha(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
alphaavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*beta(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
betaavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*gamma(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
gammaavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*v(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
vavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*w(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
wavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*cp(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
cpavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*ts(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
tsavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*tt(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
ttavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*ttr(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
ttravg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*vm(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
vmavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*wm(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
wmavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*va(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)

```

```

vaavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*wa(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
waavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*vr(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
vrag(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*wr(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
wrag(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*wc(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
wcavg(i)=num/den
temp(1:n)=radius(1:n,i)*rho(1:n,i)*va(1:n,i)*vc(1:n,i)
call integ(n,radius(1:n,i),temp(1:n),num)
vcavg(i)=num/den

mdot(i)=den*2*pi

psmax(i)=maxval(ps(1:n,i))
ptmax(i)=maxval(pt(1:n,i))
ptrmax(i)=maxval(ptr(1:n,i))
end do
ps3=psavg(3)
ps4=psavg(4)
ps5=psavg(5)
t3=tsavg(3)
t4=tsavg(4)
t5=tsavg(5)
cpa=cpavg(1)
t4s=t3*(ps4/ps3)**pwr
t5s=t4*(ps5/ps4)**pwr
t5s1=t3*(ps5/ps3)**pwr
dhs1=cpa*(t3-t4s)
dhs2=cpa*(t4-t5s)
dhs3=cpa*(t3-t5s1)
outdir=trim('out_')//trim(adjustl(rpms))//trim('/')
outdir=trim(adjustl(outdir))
floss=trim('loss_')//trim(adjustl(rpms))//trim('.dat')
floss=trim(adjustl(floss))

feff=trim('eff_')//trim(adjustl(rpms))//trim('.dat')
creact=trim('reaction_')//trim(adjustl(rpms))//trim('.dat')
floss=trim(outdir)//trim(adjustl(floss))
feff=trim(outdir)//trim(adjustl(feff))
creact=trim(outdir)//trim(adjustl(creact))
open(unit=212,file=floss,status='unknown')

WRITE(212,*)'TITLE="Loss Summary"'
WRITE(212,*)'VARIABLES="Span Stator",'&
, 'Stator Total Pressure Loss Coefficient','&
, 'Span Rotor','&
, 'Rotor Total Pressure Loss Coefficient'

OPEN(UNIT=213,FILE=FEFF,status='unknown')
WRITE(213,*)'TITLE="Efficiency Summary"'
WRITE(213,*)'VARIABLES="Span Stator",'&
, 'Stator Efficiency','&
, 'Span Rotor','&
, 'Rotor Efficiency','&
, 'Span Stage','&
, 'Stage(2) Efficiency'

open(unit=214,file=creact,status='unknown')
write(214,*)'Title="Rotor Reaction"'

superstring='VARIABLES="Span R1","Reaction R1","Stage 1 Flow Coefficient","Stage 1 Load Coefficient",'&

```

```

//'"Span R2","Reaction R2","Stage 2 Flow Coefficient","Stage 2 Load Coefficient"
write(214,'(A180)')adjustl(superstring)

do i=1,n
    ss(i)=0.5*(span(i,3)+span(i,4))
    sr(i)=0.5*(span(i,4)+span(i,5))
    sst(i)=0.5*(span(i,3)+span(i,5))
    eta_s(i)=(0.5*v(i,4)**2)/(dhs1+0.5*v(i,3)**2)
    eta_r(i)=(0.5*(w(i,5)**2-u(i,5)**2))/(dhs2+0.5*(w(i,4)**2-&
    u(i,4)**2))
    eta_st(i)=(0.5*(w(i,5)**2-u(i,5)**2+v(i,4)**2))/(dhs3+0.5*&
    (v(i,3)**2+w(i,4)**2-u(i,4)**2))
    eta_sti(i)=(1-(tt(i,5)/tt(i,3)))/(1-(ps(i,5)/pt(i,3))**pwr)
    z_s(i)=(pt(i,3)-pt(i,4))/(ptavg(3)-psavg(4))
    z_r(i)=(ptr(i,4)-ptr(i,5))/(ptravg(4)-psavg(5))
    react(i,1)=(hs(i,2)-hs(i,3))/(hs(i,1)-hs(i,3))
    react(i,2)=(hs(i,4)-hs(i,5))/(hs(i,3)-hs(i,5))
    st_flow_coeff(i,1)=vm(i,3)/u(i,3)
    st_flow_coeff(i,2)=vm(i,5)/u(i,5)
    lm=u(i,2)*vc(i,2)+u(i,3)*vc(i,3)
    st_load_coeff(i,1)=lm/(u(i,3)*u(i,3))
    lm=u(i,4)*vc(i,4)+u(i,5)*vc(i,5)
    st_load_coeff(i,2)=lm/(u(i,5)*u(i,5))
    write(212,1000)ss(i),z_s(i),sr(i),z_r(i)
    write(213,1001)ss(i),eta_s(i),sr(i),eta_r(i),sst(i),eta_st(i)
    write(214,1003)span(i,3),react(i,1),st_flow_coeff(i,1),st_load_coeff(i,1),&
    span(i,5),react(i,2),st_flow_coeff(i,2),st_load_coeff(i,2)

end do
close(212)
close(213)
close(214)
FLOSS=TRIM('loss1_')//TRIM(RPMS)//TRIM('.dat')
FLOSS=TRIM(ADJUSTL(FLOSS))
FLOSS=TRIM(outdir)//TRIM(ADJUSTL(FLOSS))
open(unit=215,file=floss,status='unknown')
write(215,*)'TITLE="Loss ver1"'
write(215,*)'VARIABLES="Normalized Span","Total Pressure Loss Coefficient <greek>z</greek>"'
write(215,*)'ZONE T="Stator RANS"'
do i=1,n
    write(215,1005)ss(i),z_s(i)
end do
write(215,*)'ZONE T="Rotor RANS"'
do i=1,n
    write(215,1005)sr(i),z_r(i)
end do
close(215)
FEFF=TRIM('Eff1_')//TRIM(RPMS)//TRIM('.dat')
FEFF=TRIM(outdir)//TRIM(ADJUSTL(FEFF))
open(unit=215,file=feff,status='unknown')
write(215,*)'TITLE="Loss ver1"'
write(215,*)'VARIABLES="Normalized Span","Efficiency <greek>h</greek>"'
write(215,*)'ZONE T="Stator RANS"'
do i=1,n
    write(215,1005)ss(i),eta_s(i)
end do
write(215,*)'ZONE T="Rotor RANS"'
do i=1,n
    write(215,1005)sr(i),eta_r(i)
end do
write(215,*)'ZONE T="Stage RANS"'
do i=1,n
    write(215,1005)sst(i),eta_st(i)
end do
close(215)

```

```

1000  format(4(f8.5,1x))
1001  format(6(f8.5,1x))
1003  format(8(f8.5,1x))
1005  format(2(f8.5,1x))
call integ(n,ss(1:n),z_s(1:n),z_s_avg)
call integ(n,sr(1:n),z_r(1:n),z_r_avg)
call integ(n,ss(1:n),eta_s(1:n),eta_s_avg)
call integ(n,sr(1:n),eta_r(1:n),eta_r_avg)
call integ(n,sst(1:n),eta_st(1:n),eta_st_avg)
call integ(n,sst(1:n),eta_sti(1:n),eta_sti_avg)

psin=psavg(1)
psout=psavg(ns)

ptin=ptavg(1)
ptout=ptavg(ns)
ttout=ttavg(ns)
ttin=ttavg(1)
pr=psavg(1)/psavg(ns)
eta_ts=(1-(ttout/ttin))/(1-((psout/ptin)**pwr))
eta_tt=(1-(ttout/ttin))/(1-((ptout/ptin)**pwr))
co=sqrt(2*1000.0*(htavg(1)-htavg(ns))/float(nstages))
u_co=umean/co
FEFF=TRIM(outdir)//TRIM('Eff_Summary_')//TRIM(RPMS)//TRIM('.dat')
FEFF=TRIM(ADJUSTL(FEFF))
OPEN(UNIT=214,FILE=FEFF,STATUS='UNKNOWN')
write(214,'(A30,1x,F10.5)')'RPM= ',RPM
write(214,'(A30,1x,F10.5)')'Pressure Ratio= ',PR
write(214,'(A30,1x,F10.5)')'massflow at exit=',mdot(ns)
write(214,'(A30,1x,F10.5)')'U/Co= ',U_CO
write(214,'(A30,1x,F10.5)')'Machine Efficiency t-s= ',ETA_TS
write(214,'(A30,1x,F10.5)')'Machine Efficiency t-t= ',ETA_TT
write(214,'(A30,1x,F10.5)')'Stator 2 Efficiency t-s= ',ETA_S_AVG
write(214,'(A30,1x,F10.5)')'Rotor 2 Efficiency= ',ETA_R_AVG
write(214,'(A30,1x,F10.5)')'Stage 2 Efficiency= ',ETA_ST_AVG
write(214,'(A30,1x,F10.5)')'Stator 2 Loss Coefficient= ',Z_S_AVG
write(214,'(A30,1x,F10.5)')'Rotor 2 Loss Coefficient= ',Z_R_AVG
write(214,'(A30,1x,F10.5)')'Rotor 1 Reaction= ',(hsavg(2)-hsavg(3))/(hsavg(1)-hsavg(3))
write(214,'(A30,1x,F10.5)')'Rotor 2 Reaction= ',(hsavg(4)-hsavg(5))/(hsavg(3)-hsavg(5))
write(214,'(A30,1x,F10.5)')'Stage 1 Flow Coefficient= ',vmavg(3)/umean
write(214,'(A30,1x,F10.5)')'Stage 2 Flow Coefficient= ',vmavg(5)/umean
lm=Umean*VCavg(2)+Umean*VCavg(3)
write(214,'(A30,1x,F10.5)')'Stage 1 Load Coefficient= ',lm/(umean*umean)
lm=Umean*VCavg(4)+Umean*VCavg(5)
write(214,'(A30,1x,F10.5)')'Stage 2 Load Coefficient= ',lm/(umean*umean)

!WRITE(214,'(8(F10.5,1x))')RPM,PR,U_CO,ETA_TS,ETA_S_AVG,ETA_R_AVG&
!      ,ETA_ST_AVG,Z_S_AVG,Z_R_AVG
CLOSE(214)

FEFF=TRIM(outdir)//'Eff_Dump_'//TRIM(RPMS)//'.dat'
OPEN(UNIT=215,FILE=FEFF,Status='unknown')
WRITE(215,1002)'Rotation Speed [1/min]= ',RPM
WRITE(215,1002)'Avg Stator Total Pressure Loss Coefficient= ',&
Z_S_AVG
WRITE(215,1002)'Avg Rotor Total Pressure Loss Coefficient= ',&
Z_R_AVG
WRITE(215,1002)'Avg Stator Efficiency= ',ETA_S_AVG
WRITE(215,1002)'Avg Rotor Efficiency= ',ETA_R_AVG
WRITE(215,1002)'Avg Stage(2) Efficiency= ',ETA_ST_AVG
WRITE(215,1002)'Avg Stage Isentropic Eff= ',ETA_STI_AVG
WRITE(215,1002)'U_mean = ',UMEAN
WRITE(215,1002)'Co = ',CO
WRITE(215,1002)'U/Co= ',U_CO

CLOSE(215)

```

```
1002 FORMAT(A45,1x,F10.5)
```

```
!c normalize pressure if needed
!call velavg(va,radius,rho,vavg)
!temp=radius(1:n,3)
!ctemp1=va(1:n,3)
!ccall integ(radius(1:n,4),radius(1:n,4)*rho(1:n,4)*va(1:n,4)*
!c      + v(1:n,4),dvva)
!ccall integ(radius(1:n,4),radius(1:n,4)*rho(1:n,4)*va(1:n,4),dva)
!cdvva=2*pi*dvva
!dva=2*pi*dva
!write(222,*)dvva
!cwrite(222,*)dva
!cdm=dvva/dva
!cwrite(222,*)dm
!cwrite(*,*)"dm= ",dm
!cwrite(222,*),dm
dum_string=trim(outdir)//trim('avg.dat')
dum_string=trim(adjustl(dum_string))
open(unit=222,file=dum_string,status='unknown')
write(222,122)(psavg(i),i=1,ns)
write(222,122)(ptavg(i),i=1,ns)
write(222,122)(ptravg(i),i=1,ns)
write(222,122)(tsavg(i),i=1,ns)
write(222,122)(ttavg(i),i=1,ns)
write(222,122)(savg(i),i=1,ns)
write(222,122)(hsavg(i),i=1,ns)
write(222,122)(htavg(i),i=1,ns)
write(222,122)(htravg(i),i=1,ns)
write(222,122)(wavg(i),i=1,ns)
write(222,122)(wmavg(i),i=1,ns)
write(222,122)(waavg(i),i=1,ns)
write(222,122)(wragv(i),i=1,ns)
write(222,122)(vavg(i),i=1,ns)
write(222,122)(vmavg(i),i=1,ns)
write(222,122)(vaavg(i),i=1,ns)
write(222,122)(vravg(i),i=1,ns)
write(222,122)(alphaavg(i)*r2d,i=1,ns)
write(222,122)(betaavg(i)*r2d,i=1,ns)
write(222,122)(gammaavg(i)*r2d,i=1,ns)

121 format(A110)
122 format (5(F20.8,2x))
123 format (A1500)
FOUT1=TRIM(outdir)//TRIM('cfplot_')//TRIM(RPMS)//TRIM('.dat')
FOUT2=TRIM(outdir)//TRIM('cfplot_full_')//TRIM(RPMS)//TRIM('.dat')
FOUT1=TRIM(ADJUSTL(FOUT1))
FOUT2=TRIM(ADJUSTL(FOUT2))
call get_supstring(ver,norm,superstring)
write(14,123)adjustl(superstring)
do j=1,ns
    if(j==1)write(14,*)'ZONE T="Station 1 CFD-RANS"'
    if(j==2)write(14,*)'ZONE T="Station 2 CFD-RANS"'
    if(j==3)write(14,*)'ZONE T="Station 3 CFD-RANS"'
    if(j==4)write(14,*)'ZONE T="Station 4 CFD-RANS"'
    if(j==5)write(14,*)'ZONE T="Station 5 CFD-RANS"'
    if(j==6)write(14,*)'ZONE T="Station 6 CFD-RANS"'
    if(j==7)write(14,*)'ZONE T="Station 7 CFD-RANS"'
    do i=1,n
        write(14,1978)span(i,j),pt(i,j),ptr(i,j),ps(i,j),v(i,j),&
        vc(i,j),vr(i,j),va(i,j),w(i,j),wc(i,j),wr(i,j),wa(i,j),u(i,j),&
        gamma1(i,j)*180.0/pi,alpha(i,j)*180.0/pi,beta(i,j)*180.0/pi,&
        mm(i,j),mmr(i,j),ts(i,j),tt(i,j),ttr(i,j),&
        cp(i,j),s(i,j),hs(i,j),ht(i,j),htr(i,j),rho(i,j),visc(i,j),&
        rc(i,j),gamma(i,j)*180.0/pi,tke(i,j)
```

```

end do
close(14)

open(unit=14,file=fout1,status='unknown')
call get_supstring(ver,norm,superstring)
write(14,123)adjustl(superstring)
do j=3,5
    if(j==3)write(14,*)'ZONE T="Station 3 CFD-RANS"'
    if(j==4)write(14,*)'ZONE T="Station 4 CFD-RANS"'
    if(j==5)write(14,*)'ZONE T="Station 5 CFD-RANS"'
    do i=1,n
        if (norm == 0)then
            ptn=pt(i,j);ptrn=ptr(i,j);psn=ps(i,j)
        elseif(norm==1)then
            psn=(ps(i,j)-psmin(j))/(psmax(j)-psmin(j))
            ptn=(pt(i,j)-ptmin(j))/(ptmax(j)-ptmin(j))
            ptrn=(ptr(i,j)-ptrmin(j))/(ptrmax(j)-ptrmin(j))
        elseif(norm==2)then
            psn=ps(i,j)/psavg(3)
            ptn=pt(i,j)/ptavg(3)
            ptrn=ptr(i,j)/ptravg(3)
        elseif(norm==3)then
            psn=ps(i,j)/psmax(3)
            ptn=pt(i,j)/ptmax(3)
            ptrn=ptr(i,j)/ptrmax(3)
        end if
        write(14,1978)span(i,j),ptn,ptrn,psn,v(i,j),&
            vc(i,j),vr(i,j),va(i,j),w(i,j),wc(i,j),wr(i,j),wa(i,j),u(i,j),&
            gamma1(i,j)*180.0/pi,alpha(i,j)*180.0/pi,beta(i,j)*180.0/pi&
            ,mm(i,j),mmr(i,j),ts(i,j),tt(i,j),ttr(i,j),&
            cp(i,j),s(i,j),hs(i,j),ht(i,j),htr(i,j),rho(i,j),visc(i,j),&
            rc(i,j),gamma(i,j)*180.0/pi,tke(i,j)
    end do
end do

1978 format(27(f15.5,1x),e15.5,1x,3(f15.5,1x))
19 format(30(a86,1x))

deallocate(temp,temp1,stat=iq)
deallocate(z_s,z_r,eta_s,eta_r,stat=iq)
deallocate(eta_st,eta_sti,ss,sr,sst,stat=iq)
deallocate(st_flow_coeff,st_load_coeff,stat=iq)
deallocate(react,stat=iq)
deallocate(span,pt,ps,ptr,mm,mmr,stat=iq)
deallocate(ts,tt,ttr,rho,visc,rc,stat=iq)
deallocate(cp,s,hs,hi,ht,htr,ro,stat=iq)
deallocate(u,v,vu,vv,vw,va,vc,stat=iq)
deallocate(vr,vm,w,wu,wv,ww,wa,wc,wr,wm,stat=iq)
deallocate(tke,tet,ted,eddyv,radius,stat=iq)
deallocate(alpha,beta,gamma,vb2b,wb2b,stat=iq)
deallocate(c,cc,wdist,stat=iq)

end

subroutine contour_analysis(nx,ns,nst,nxc)
use parametr
use turbine
use output_dir
use pavg

integer::nx,ns,nst,iq,nn,ii
integer,dimension(50)::nxc,cno
character(len=80)::s1,s2,s3,s4,fout
character(len=5)::is

```



```

character(len=1500)::superstring
real(8)::dum,psn,ptn,ptrn
real(8),allocatable,dimension(:,:)::ps,pt,ptr,s,hs,ht,htr,hi
real(8),allocatable,dimension(:,:)::ts,tt,ttr,ro,vb2b,wb2b
real(8),allocatable,dimension(:,:)::y,z,rc,cp,u,v,vu,vv,vw,vm
real(8),allocatable,dimension(:,:)::va,vc,vr,w,wu,wv,ww,wm,wa,wc,wr,uu
real(8),allocatable,dimension(:,:)::alpha,beta,gamma,gamma1,mm,mmr,c,rho,visc,tke
real(8),allocatable,dimension(:,:)::ted,tef,eddyv,radius,angle,angleo
real(8),allocatable,dimension(:)::sector
real(8),dimension(nst)::psmax,ptmax,ptrmax,psmin,ptmin,ptrmin

s1='contV_'
s2='contP_'
write(*,*)'allocating memory size',nx

allocate(sector(nbr),stat=iq)
allocate(y(nx,nst),z(nx,nst),u(nx,nst),v(nx,nst),vu(nx,nst),vv(nx,nst),stat=iq)
allocate(vw(nx,nst),va(nx,nst),vc(nx,nst),vr(nx,nst),vm(nx,nst),w(nx,nst),stat=iq)
allocate(wu(nx,nst),wv(nx,nst),ww(nx,nst),wa(nx,nst),wc(nx,nst),wr(nx,nst),stat=iq)
allocate(wm(nx,nst),alpha(nx,nst),beta(nx,nst),gamma(nx,nst),gamma1(nx,nst),mm(nx,nst),stat=iq)
allocate(mmr(nx,nst),c(nx,nst),tke(nx,nst),tef(nx,nst),ted(nx,nst),eddyv(nx,nst),stat=iq)
allocate(rho(nx,nst),visc(nx,nst),ps(nx,nst),pt(nx,nst),ptr(nx,nst),stat=iq)
allocate(ts(nx,nst),tt(nx,nst),ttr(nx,nst),rc(nx,nst),cp(nx,nst),s(nx,nst),stat=iq)
allocate(hs(nx,nst),hi(nx,nst),ht(nx,nst),htr(nx,nst),ro(nx,nst),vb2b(nx,nst),wb2b(nx,nst),stat=iq)
allocate(radius(nx,nst),angle(nx,nst),angleo(nx,nst),stat=iq)
if (iq/=0)then
    write(*,*)'Issues allocating memory'
    stop
end if

do i=1,nstages*2
    sector(i)=2*pi/nblades(i)
end do

do i=1,nst
    ii=i+2
    write(is,'(I2)')ii
    s3=trim(s1)//trim(adjustl(is))//'.dat'
    s4=trim(s2)//trim(adjustl(is))//'.dat'
    cno(i)=0
    open(unit=3,file=s3,status='old')
    open(unit=4,file=s4,status='old')
    do j=1,nxc(i)
        read(3,*)y(j,i),z(j,i),u(j,i),v(j,i),vu(j,i),vv(j,i),vw(j,i),&
        va(j,i),vc(j,i),vr(j,i),vm(j,i),w(j,i),wu(j,i),wv(j,i),&
        ww(j,i),wa(j,i),wc(j,i),wr(j,i),wm(j,i),alpha(j,i),beta(j,i),&
        gamma(j,i),mm(j,i),mmr(j,i),c(j,i),tke(j,i),ted(j,i),tef(j,i),&
        eddyv(j,i),rho(j,i),visc(j,i)

        read(4,*)dum,dum,ps(j,i),pt(j,i),ptr(j,i),ts(j,i),tt(j,i),&
        ttr(j,i),rc(j,i),cp(j,i),s(j,i),hs(j,i),hi(j,i),ht(j,i),&
        htr(j,i),ro(j,i),vb2b(j,i),wb2b(j,i)

        vb2b(j,i)=sqrt(va(j,i)**2+vc(j,i)**2)
        alpha(j,i)=acos(vc(j,i)/vb2b(j,i))
        gamma(j,i)=atan(vr(j,i)/va(j,i))
        gamma1(j,i)=asin(vr(j,i)/v(j,i))

        radius(j,i)=sqrt(y(j,i)**2+z(j,i)**2)
        u(j,i)=omega*radius(j,i)

        wc(j,i)=vc(j,i)-u(j,i)
        wa(j,i)=va(j,i)
        wr(j,i)=vr(j,i)

        wb2b(j,i)=sqrt(wa(j,i)**2+wc(j,i)**2)

```

```

w(j,i)=sqrt(wb2b(j,i)**2+wr(j,i)**2)
vvec_i=wb2b(j,i)*cos(alpha(j,i))

beta(j,i)=acos(wc(j,i)/wb2b(j,i))

pt(j,i)=ps(j,i)+0.5*rho(j,i)*(v(j,i)*v(j,i))
ptr(j,i)=ps(j,i)+0.5*rho(j,i)*(w(j,i)*w(j,i)-u(j,i)*u(j,i))

ht(j,i)=hs(j,i)+0.5*v(j,i)*v(j,i)
htr(j,i)=hs(j,i)+0.5*(w(j,i)*w(j,i)-u(j,i)*u(j,i))

c(j,i)=v(j,i)/mm(j,i)
mmr(j,i)=w(j,i)/c(j,i)
angle(j,i)=atan(y(j,i)/z(j,i))

end do
close(3)
close(4)

end do

angleo=angle
ps=ps+pcor;pt=pt+pcor;ptr=ptr+pcor
ps=ps/1000.;ptr=ptr/1000.;pt=pt/1000.
s=s/1000.;hs=hs/1000;ht=ht/1000;htr=htr/1000.
do j=1,nst
    psmax(j)=maxval(ps(1:nxc(j),j))
    psmin(j)=minval(ps(1:nxc(j),j))
    ptmax(j)=maxval(pt(1:nxc(j),j))
    ptmin(j)=minval(pt(1:nxc(j),j))
    ptrmax(j)=maxval(ptr(1:nxc(j),j))
    ptrmin(j)=minval(ptr(1:nxc(j),j))

end do

fout=trim(outdir)//trim('contour')//trim(rpms)//trim('.dat')
fout=trim(adjustl(fout))
call get_supstringc(ver,norm,superstring)

open(unit=14,file=fout,status='unknown')
write(14,'(A1500)')superstring
do j=1,nst
    jj=j+2-1

    if(j == 1)write(14,*)'ZONE T="Station 3 CFD-RANS"'
    if(j == 2)write(14,*)'ZONE T="Station 4 CFD-RANS"'
    if(j == 3)write(14,*)'ZONE T="Station 5 CFD-RANS"'

do k=-1,1

do i=1,nxc(j)

if (norm == 0)then
ptn=pt(i,j);ptrn=ptr(i,j);psn=ps(i,j)
elseif(norm==1)then
psn=(ps(i,j)-psmin(j))/(psmax(j)-psmin(j))
ptn=(pt(i,j)-ptmin(j))/(ptmax(j)-ptmin(j))
ptrn=(ptr(i,j)-ptrmin(j))/(ptrmax(j)-ptrmin(j))
elseif(norm==2)then
psn=ps(i,j)/psavg(3)
ptn=pt(i,j)/ptavg(3)
ptrn=ptr(i,j)/ptravg(3)
elseif(norm==3)then
psn=ps(i,j)/psmax(3)
ptn=pt(i,j)/ptmax(3)

```

```

ptrn=ptr(i,j)/ptrmax(3)
end if

angle(i,j)=angleo(i,j)+float(k)*sector(jj)
y(i,j)=radius(i,j)*sin(angle(i,j))
z(i,j)=radius(i,j)*cos(angle(i,j))
write(14,1000)y(i,j),z(i,j),ptn,ptrn,psn,v(i,j),&
vc(i,j),vr(i,j),va(i,j),w(i,j),wc(i,j),wr(i,j),&
wa(i,j),u(i,j),gamma1(i,j)*r2d,alpha(i,j)*r2d,&
beta(i,j)*r2d,mm(i,j),mmr(i,j),ts(i,j),tt(i,j),&
ttr(i,j),cp(i,j),s(i,j),hs(i,j),ht(i,j),htr(i,j),&
rho(i,j),visc(i,j),rc(i,j),gamma(i,j)*r2d,tcd(i,j),&
tke(i,j),tef(i,j),eddyv(i,j),ro(i,j),angle(i,j)*r2d

end do

end do
end do

close(14)
1000 format(31(F15.5),1x,4(E15.5),1x,F15.5,1x,F15.5)
end subroutine

subroutine get_supstring(ver,norm,string)
integer::norm,ver
character(len=1500)::string
!*****
!norm=0 no normalization
!norm=1 in station max and min values
!norm=2 Station 3 average values
!norm=3 Station 3 maximum values
!ver =0 signifies newer version of tecplot format
!ver !=0 signifies older version of tecplot format
!*****

if (ver==0)then
  if (norm==0)then
    string='VARIABLES="Normalized Span r<sup>*</sup>",&
    //'Total Pressure [kPa],"Relative Total Pressure [kPa],"&
    //'Static Pressure [kPa],"Absolute Velocity [m/sec],"&
    //'Absolute Circumferential Velocity [m/sec],"&
    //'Absolute Radial Velocity [m/sec],"&
    //'Absolute Axial Velocity [m/sec],"&
    //'Relative Velocity [m/sec],"&
    //'Relative Circumferential Velocity [m/sec],"&
    //'Relative Radial Velocity [m/sec],"&
    //'Relative Axial Velocity [m/sec],"&
    //'Rotational Velocity [m/sec],"&
    //'<greek>g</greek> compound [deg],"<greek>a</greek> [deg],"<greek>b</greek> [deg],"&
    //'Mach Number","Relative Mach Number",&
    //'Static Temperature [K],"&
    //'Total Temperature [K],"&
    //'Relative Total Temperature [K],"&
    //'Specific Heat Capacity-C<sub>p</sub> [kJ/kg/K],"&
    //'Entropy [kJ/kg/K],"Enthalpy [kJ/kg],"&
    //'Total Enthalpy [kJ/kg],"&
    //'Relative Total Enthalpy [kJ/kg],"&
    //'Density-<greek>r</greek> [kg/m<sup>3</sup>],"&
    //'Dynamic Viscosity-<greek>m</greek> [kg/sec/m],"&
    //'Gas Constant-R [J/kg/K],"&
    //'<greek>g</greek> [deg],"&
    //'TKE [m<sup>2</sup>/sec<sup>2</sup>]"'
  else
    string='VARIABLES="Normalized Span r<sup>*</sup>",&
    //'Normalized Total Pressure","Normalized Relative Total Pressure",&
    //'Normalized Static Pressure","Absolute Velocity [m/sec],"&

```

```

//'"Absolute Circumferential Velocity [m/sec]",'&
//'"Absolute Radial Velocity [m/sec]",'&
//'"Absolute Axial Velocity [m/sec]",'&
//'"Relative Velocity [m/sec]",'&
//'"Relative Circumferential Velocity [m/sec]",'&
//'"Relative Radial Velocity [m/sec]",'&
//'"Relative Axial Velocity [m/sec]",'&
//'"Rotational Velocity [m/sec]",'&
//'"<greek>g</greek> compound [deg]", "<greek>a</greek> [deg]", "<greek>b</greek> [deg]",'&
//'"Mach Number", "Relative Mach Number", ' &
//'"Static Temperature [K]",'&
//'"Total Temperature [K]",'&
//'"Relative Total Temperature [K]",'&
//'"Specific Heat Capacity-C<sub>p</sub> [J/kg/K]",'&
//'"Entropy [kJ/kg/K]", "Enthalpy [kJ/kg]",'&
//'"Total Enthalpy [kJ/kg]",'&
//'"Relative Total Enthalpy [kJ/kg]",'&
//'"Density-<greek>r</greek> [kg/m<sup>3</sup>]",' &
//'"Dynamic Viscosity-<greek>m</greek> (kg/sec/m)",'&
//'"Gas Constant-R [J/kg/K]",'&
//'"<greek>g</greek> [deg]",'&
//'"TKE [m<sup>2</sup>/sec<sup>2</sup>]"',

    end if

elseif(ver==1)then
    if(norm==0)then
        string='VARIABLES="Normalized Span r*",'&
//'"Total Pressure [kPa]", "Relative Total Pressure [kPa]",'&
//'"Static Pressure [kPa]", "Absolute Velocity [m/sec]",'&
//'"Absolute Circumferential Velocity [m/sec]",'&
//'"Absolute Radial Velocity [m/sec]",'&
//'"Absolute Axial Velocity [m/sec]",'&
//'"Relative Velocity [m/sec]",'&
//'"Relative Circumferential Velocity [m/sec]",'&
//'"Relative Radial Velocity [m/sec]",'&
//'"Relative Axial Velocity [m/sec]",'&
//'"Rotational Velocity [m/sec]",'&
//'"'g compound [deg]", "'a [deg]", "'b [deg]",'&
//'"Mach Number", "Relative Mach Number", ' &
//'"Static Temperature [K]",'&
//'"Total Temperature [K]",'&
//'"Relative Total Temperature [K]",'&
//'"Specific Heat Capacity-Cp [J/kg/K]",'&
//'"Entropy [kJ/kg/K]", "Enthalpy [kJ/kg]",'&
//'"Total Enthalpy [kJ/kg]",'&
//'"Relative Total Enthalpy [kJ/kg]",'&
//'"Density-<greek>r</greek> [kg/m^3]",' &
//'"Dynamic Viscosity-'m [kg/sec/m]",'&
//'"Gas Constant-R [J/kg/K]",'&
//'"'g [deg]",'&
//'"TKE [m^2/sec^2]"',

    else
        string='VARIABLES="Normalized Span r*",'&
//'"Normalized Total Pressure", "Normalized Relative Total Pressure", ' &
//'"Normalized Static Pressure", "Absolute Velocity [m/sec]",'&
//'"Absolute Circumferential Velocity [m/sec]",'&
//'"Absolute Radial Velocity [m/sec]",'&
//'"Absolute Axial Velocity [m/sec]",'&
//'"Relative Velocity [m/sec]",'&
//'"Relative Circumferential Velocity [m/sec]",'&
//'"Relative Radial Velocity [m/sec]",'&
//'"Relative Axial Velocity [m/sec]",'&
//'"Rotational Velocity [m/sec]",'&
//'"'g compound [deg]", "'a [deg]", "'b [deg]",'&
//'"Mach Number", "Relative Mach Number", ' &

```

```

        //'Static Temperature [K]",&
        //'Total Temperature [K]",&
        //'Relative Total Temperature [K]",&
        //'Specific Heat Capacity-Cp [J/kg/K]",&
        //'Entropy [kJ/kg/K]","Enthalpy [kJ/kg]",&
        //'Total Enthalpy [kJ/kg]",&
        //'Relative Total Enthalpy [kJ/kg]",&
        //'Density-<greek>r</greek> [kg/m^3]",&
        //'Dynamic Viscosity-'m [kg/sec/m]",&
        //'Gas Constant-R [J/kg/K]",&
        //'g [deg]",&
        //'TKE [m^2/sec^2]"
    endif

else
    write(*,*)'header error'
    string=' '
    return
end if

end subroutine
!*****
subroutine integ(n,x,fx,valinteg)
implicit none
integer :: n,i
real(8),dimension(n) :: x,fx
real(8) :: sum,valinteg

sum=0.0

do i=1,n-1
    sum=sum+(x(i+1)-x(i))*0.5*(fx(i)+fx(i+1))
end do
valinteg=sum
return
end

!*****

subroutine get_supstringc(ver,norm,string)
implicit none
integer::norm,ver
character(len=1500)::string
!*****
!norm=0 no normalization
!norm=1 in station max and min values
!norm=2 Station 3 average values
!norm=3 Station 3 maximum values
!ver =0 signifies newer version of tecplot format
!ver !=0 signifies older version of tecplot format
!*****

if (ver==0)then
    if(norm==0)then
        string='VARIABLES="Y","Z",&
        //'Total Pressure [kPa]",&
        //'Relative Total Pressure [kPa]",&
        //'Static Pressure [kPa]",&
        //'Absolute Velocity [m/sec]",&

```

```

//'"Absolute Circumferential Velocity [m/sec]", '&
//'"Absolute Radial Velocity [m/sec]", '&
//'"Absolute Axial Velocity [m/sec]", '&
//'"Relative Velocity [m/sec]", '&
//'"Relative Circumferential Velocity [m/sec]", '&
//'"Relative Radial Velocity [m/sec]", '&
//'"Relative Axial Velocity [m/sec]", '&
//'"Rotational Velocity [m/sec]", '&
//'"<greek>g</greek> compound [deg]", "<greek>a</greek> [deg]", "<greek>b</greek> [deg]", '&
//'"Mach Number", "Relative Mach Number", '&
//'"Static Temperature [K]", '&
//'"Total Temperature [K]", '&
//'"Relative Total Temperature [K]", '&
//'"Specific Heat Capacity - C<sub>p</sub> [J/kg/K]", '&
//'"Entropy [kJ/kg/K]", "Enthalpy [kJ/kg]", '&
//'"Total Enthalpy [kJ/kg]", '&
//'"Relative Total Enthalpy [kJ/kg]", '&
//'"Density - <greek>r</greek> [kg/m<sup>3</sup>]", '&
//'"Dynamic Viscosity-<greek>m</greek> [kg/s/m]", '&
//'"Gas Constant - R [J/kg/K]", '&
//'"<greek>g</greek> [deg]", '&
//'"Turbulence Eddy Dissipation [1/s]", '&
//'"Turbulence Kinetic Energy [m<sup>2</sup>/sec<sup>2</sup>]", '&
//'"Turbulence Eddy Frequency [1/s]", '&
//'"Eddy Viscosity [kg m/s]", '&
//'"Rothalpy[kJ/Kg]", "angle"'

else
    string='VARIABLES="Y", "Z", '&
//'"Normalized Total Pressure", '&
//'"Normalized Relative Total Pressure", '&
//'"Normalized Static Pressure", '&
//'"Absolute Velocity [m/sec]", '&
//'"Absolute Circumferential Velocity [m/sec]", '&
//'"Absolute Radial Velocity [m/sec]", '&
//'"Absolute Axial Velocity [m/sec]", '&
//'"Relative Velocity [m/sec]", '&
//'"Relative Circumferential Velocity [m/sec]", '&
//'"Relative Radial Velocity [m/sec]", '&
//'"Relative Axial Velocity [m/sec]", '&
//'"Rotational Velocity [m/sec]", '&
//'"<greek>g</greek> compound [deg]", "<greek>a</greek> [deg]", "<greek>b</greek> [deg]", '&
//'"Mach Number", "Relative Mach Number", '&
//'"Static Temperature [K]", '&
//'"Total Temperature [K]", '&
//'"Relative Total Temperature [K]", '&
//'"Specific Heat Capacity - C<sub>p</sub> [J/kg/K]", '&
//'"Entropy [kJ/kg/K]", "Enthalpy [kJ/kg]", '&
//'"Total Enthalpy [kJ/kg]", '&
//'"Relative Total Enthalpy [kJ/kg]", '&
//'"Density - <greek>r</greek> [kg/m<sup>3</sup>]", '&
//'"Dynamic Viscosity-<greek>m</greek> [kg/s/m]", '&
//'"Gas Constant - R [J/kg/K]", '&
//'"<greek>g</greek> [deg]", '&
//'"Turbulence Eddy Dissipation [1/s]", '&
//'"Turbulence Kinetic Energy [m<sup>2</sup>/sec<sup>2</sup>]", '&
//'"Turbulence Eddy Frequency [1/s]", '&
//'"Eddy Viscosity [kg m/s]", '&
//'"Rothalpy[kJ/Kg]", "angle"'

    end if

elseif(ver==1)then
    if(norm==0)then
        string='VARIABLES="Y", "Z", '&
//'"Total Pressure [kPa]", '&
//'"Relative Total Pressure [kPa]", '&

```

```

        //'Static Pressure [kPa],"&
        //'Absolute Velocity [m/sec],"&
        //'Absolute Circumferential Velocity [m/sec],"&
        //'Absolute Radial Velocity [m/sec],"&
        //'Absolute Axial Velocity [m/sec],"&
        //'Relative Velocity [m/sec],"&
        //'Relative Circumferential Velocity [m/sec],"&
        //'Relative Radial Velocity [m/sec],"&
        //'Relative Axial Velocity [m/sec],"&
        //'Rotational Velocity [m/sec],"&
        //'g compound [deg],"a [deg],"b [deg],"&
        //'Mach Number","Relative Mach Number",&
        //'Static Temperature [K],"&
        //'Total Temperature [K],"&
        //'Relative Total Temperature [K],"&
        //'Specific Heat Capacity - Cp [J/kg/K],"&
        //'Entropy [kJ/kg/K],"Enthalpy [kJ/kg],"&
        //'Total Enthalpy [kJ/kg],"&
        //'Relative Total Enthalpy [kJ/kg],"&
        //'Density - 'r [kg/m<sup>3</sup>],"&
        //'Dynamic Viscosity-'m [kg/s/m],"&
        //'Gas Constant-R [J/kg/K],"&
        //'g [deg],"&
        //'Turbulence Eddy Dissipation [1/s],"&
        //'Turbulence Kinetic Energy [m^2/sec^2],"&
        //'Turbulence Eddy Frequency [1/s],"&
        //'Eddy Viscosity [kg m/s],"&
        //'Rothalpy[kJ/Kg],"angle"

    else
        string='VARIABLES="Y","Z",&
        //'Normalized Total Pressure",&
        //'Normalized Relative Total Pressure",&
        //'Normalized Static Pressure",&
        //'Absolute Velocity [m/sec],"&
        //'Absolute Circumferential Velocity [m/sec],"&
        //'Absolute Radial Velocity [m/sec],"&
        //'Absolute Axial Velocity [m/sec],"&
        //'Relative Velocity [m/sec],"&
        //'Relative Circumferential Velocity [m/sec],"&
        //'Relative Radial Velocity [m/sec],"&
        //'Relative Axial Velocity [m/sec],"&
        //'Rotational Velocity [m/sec],"&
        //'g compound [deg],"a [deg],"b [deg],"&
        //'Mach Number","Relative Mach Number",&
        //'Static Temperature [K],"&
        //'Total Temperature [K],"&
        //'Relative Total Temperature [K],"&
        //'Specific Heat Capacity - Cp [J/kg/K],"&
        //'Entropy [kJ/kg/K],"Enthalpy [kJ/kg],"&
        //'Total Enthalpy [kJ/kg],"&
        //'Relative Total Enthalpy [kJ/kg],"&
        //'Density - 'r [kg/m<sup>3</sup>],"&
        //'Dynamic Viscosity-'m [kg/s/m],"&
        //'Gas Constant-R [J/kg/K],"&
        //'g [deg],"&
        //'Turbulence Eddy Dissipation [1/s],"&
        //'Turbulence Kinetic Energy [m^2/sec^2],"&
        //'Turbulence Eddy Frequency [1/s],"&
        //'Eddy Viscosity [kg m/s],"&
        //'Rothalpy[kJ/Kg],"angle"

    endif

else
write(*,*)'header error'
string='

```

```

return
end if

end subroutine

```

A.2 Fortran Program to Analyze URANS Based Turbine CFD Data

```

!*****
module params

real(8)::pi,d2r,r2d,rpm,pref,rhub,rtip,totaltime,omega,umean,dt
integer::i,j,k,nstages,ver
integer,dimension(20)::nb
real(8),dimension(20)::dpitch,dtheta,cycle_time,blade_pos
parameter(rhub=11.0*25.4e-3,rtip=13.5*25.4e-3)
parameter(pi=3.1415927,d2r=pi/180.0,r2d=1.0/d2r)
real(8),parameter::cp=1004.1,ka=1.4,pwr=(ka-1)/ka
character(len=20)::rpms
real(8)::pcor
logical::docont
save
end module params
!*****

!*****
module averager

real(8),dimension(500)::psavg,ptavg,ptravg,tsavg,ttavg,ttravg
real(8),dimension(500)::psmin,ptmin,ptrmin,psmax,ptmax,ptrmax
real(8),dimension(500)::hsavg,htavg,htravg,vmax,mdot,vavg
real(8)::u_co,pr,eta_ts,eta_tt,co
save
end module averager
!*****

program master_trans

use params

use averager
implicit none

integer::n,ns,ni,rpmi,ntime,c,norm,cont,nbr,c1,c2,c3
integer::c11,c22,c33
integer,dimension(20)::check,ncont,checka,nconta
character(len=80)::ctemp
character(len=3)::ans
real(8)::tstep,tstepn
real(8),dimension(20)::dst,dstn,binst

write(*,*)"Transient Analysis Code-Welcome"
write(*,*)"Author: Sherif Abdelfattah"
docont=.false.
open(unit=3,file='tinput.deck',status='old')
read(3,*)
read(3,*)nstages
read(3,*)
read(3,*)nb(1:2*nstages)
read(3,*)
read(3,*)dt !read time march step
!dt=dt*2 !not sure why this is multiplied by 2
read(3,*)
read(3,*)ntime !read number of time steps
read(3,*)
read(3,*)rpm !read turbine rpm
read(3,*)

```



```

read(3,*)norm,ver !read kind of pressure normalization and output version
read(3,*)
read(3,*)ans
if((ver>1).and.(ver <0))then
ver = 0
end if
if((norm>1).and.(norm<0))then
norm=0
end if
close(3)
write(*,*)'Enter correction to pressure'
read(*,*)pcor
if((ans(1:1)=='y').or.(ans(1:1)=='Y'))docont=.true.

ns=(nstages-1)*2+1
!nbr=nstages*2 !number of blade rows)
nbr=6

omega=rpm*2*pi/60.0 !calculate rpm in rad/s

do i=1,2*nstages
dpitch(i)=2*pi/nb(i)
dtheta(i)=omega*dt
cycle_time(i)=dpitch(i)/omega

end do
!calculating cycle time

do i=1,nbr
dtheta(i)=2*pi/nb(i) !size of segment per blade row 2*pi/blade number
dpitch(i)=dtheta(i)/omega !pitch size
binst(i)=dpitch(i)/omega !time in each blade segment
end do

totaltime=ntime*dt

ctemp='Lineplots/conv/Line_Station3_1.dat'
open(unit=3,file=ctemp,status='old')
c=0;c1=0;c2=0;c3=0;c11=0;c22=0;c33=0

do while(.not. eof(3))
read(3,*)
c=c+1
end do
close(3)

ctemp='Contours/conv/cont3_1.dat'
open(unit=3,file=ctemp,status='old')
do while(.not. eof(3))
read(3,*)
c1=c1+1
end do
close(3)

ctemp='Contours/conv/cont4_1.dat'
open(unit=3,file=ctemp,status='old')
do while(.not. eof(3))
read(3,*)
c2=c2+1
end do
close(3)

ctemp='Contours/conv/cont5_1.dat'
open(unit=3,file=ctemp,status='old')
do while(.not. eof(3))
read(3,*)

```

```

        c3=c3+1
    end do
    close(3)

    !Aux Contours*****
    ctemp='Contours/aux/cont3_1.dat'
    open(unit=3,file=ctemp,status='old')
    do while(.not. eof(3))
        read(3,*)
        c11=c11+1
    end do
    close(3)

    ctemp='Contours/aux/cont4_1.dat'
    open(unit=3,file=ctemp,status='old')
    do while(.not. eof(3))
        read(3,*)
        c22=c22+1
    end do
    close(3)

    ctemp='Contours/aux/cont5_1.dat'
    open(unit=3,file=ctemp,status='old')
    do while(.not. eof(3))
        read(3,*)
        c33=c33+1
    end do
    close(3)
    !Aux Contours*****
    check(1)=c1;check(2)=c2;check(3)=c3
    checka(1)=c11;checka(2)=c22;checka(3)=c33
    ncont=maxval(check(1:ns))
    nconta=maxval(checka(1:ns))

    !Get massflow versus time

    !Lineplot analysis
    write(*,*)'starting linedata analysis'
    call tline_analysis(c,ns-2,ntime,norm)
    write(*,*)'line data analysis complete'

    !Table Analysis
    write(*,*)'starting table analysis'
    call table_analysis(ns,ntime)
    write(*,*)'Table analysis complete'

    !Contour analysis
    write(*,*)'starting contour analysis'
    call contour_analysis(ncont,check,ns-2,ntime,norm)
    write(*,*)'contour analysis complete'

    if(docont==.true.)then
        write(*,*)'starting contour aux analysis'
        call contour_anal_aux(nconta,checka,ns-2,ntime,norm)
        write(*,*)'contour aux analysis complete'
    end if
    write(*,*)'ALL DONE'

end program

!*****
real(8) function integ(n,x,fx)
implicit none
integer :: n,i
real(8),dimension(n) :: x,fx

```

```

real(8) :: sum

sum=0.0

do i=1,n-1
    sum=sum+(x(i+1)-x(i))*0.5*(fx(i)+fx(i+1))
end do
integ=sum
return
end

!*****

!call make_gnu_line(n,ns,outfile,sp,psa)
subroutine make_gnu_line(n,ns,filen,x,y)

real(8),dimension(n,ns)::x,y
integer::n,ns
character(len=80)::filen

open(unit=11,file=filen,status='unknown')

do i=1,n
    write(11,50)x(i,1),y(i,1),x(i,2),y(i,2),x(i,3),y(i,3)
end do

close(11)
50 format(6(F20.8,1x))

end subroutine

subroutine table_analysis(ns,n)
use params

implicit none

integer::n,ns,iq

!allocate memory for table based values
real(8),allocatable,dimension(:)::H1,H3,H4,H5,H6
real(8),allocatable,dimension(:)::Ht1,Ht3,Ht4,Ht5,Ht6
real(8),allocatable,dimension(:)::Htr1,Htr3,Htr4,Htr5,Htr6,PRatio
real(8),allocatable,dimension(:)::Ps1,Ps3,Ps4,Ps5,Ps6
real(8),allocatable,dimension(:)::Pt1,Pt3,Pt4,Pt5,Pt6
real(8),allocatable,dimension(:)::Ptr1,Ptr3,Ptr4,Ptr5,Ptr6
real(8),allocatable,dimension(:)::T1,T3,T4,T5,T6
real(8),allocatable,dimension(:)::Tt1,Tt3,Tt4,Tt5,Tt6
real(8),allocatable,dimension(:)::V1,V3,V4,V5,V6
real(8),allocatable,dimension(:)::W1,W3,W4,W5,W6
real(8),allocatable,dimension(:)::zs,zr,ett,ets,estage,mext

real(8)::H1a,H3a,H4a,H5a,H6a
real(8)::Ht1a,Ht3a,Ht4a,Ht5a,Ht6a
real(8)::Htr1a,Htr3a,Htr4a,Htr5a,Htr6a,PRa
real(8)::Ps1a,Ps3a,Ps4a,Ps5a,Ps6a
real(8)::Pt1a,Pt3a,Pt4a,Pt5a,Pt6a
real(8)::Ptr1a,Ptr3a,Ptr4a,Ptr5a,Ptr6a
real(8)::T1a,T3a,T4a,T5a,T6a
real(8)::Tt1a,Tt3a,Tt4a,Tt5a,Tt6a
real(8)::V1a,V3a,V4a,V5a,V6a
real(8)::W1a,W3a,W4a,W5a,W6a
real(8)::zsa,zra,etta,etsa,estagea,mexta
real(8)::Pdatum

```

```

real(8)::mult,dtme

character(len=80)::indir,filein
character(len=4)::ext
character(len=6)::noint
indir='Table_Values/avg_values'
ext='.txt'

mult=1000.
allocate(pratio(n),stat=iq)
allocate(h1(n),h3(n),h4(n),h5(n),h6(n),stat=iq)
allocate(ht1(n),ht3(n),ht4(n),ht5(n),ht6(n),stat=iq)
allocate(htr1(n),htr3(n),htr4(n),htr5(n),htr6(n),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"
allocate(ps1(n),ps3(n),ps4(n),ps5(n),ps6(n),stat=iq)
allocate(pt1(n),pt3(n),pt4(n),pt5(n),pt6(n),stat=iq)
allocate(ptr1(n),ptr3(n),ptr4(n),ptr5(n),ptr6(n),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"
allocate(t1(n),t3(n),t4(n),t5(n),t6(n),stat=iq)
allocate(tt1(n),tt3(n),tt4(n),tt5(n),tt6(n),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"
allocate(V1(n),V3(n),V4(n),V5(n),V6(n),stat=iq)
allocate(W1(n),W3(n),W4(n),W5(n),W6(n),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"
allocate(zs(n),zr(n),ett(n),ets(n),estage(n),mext(n),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"
H1a=0.;H3a=0.;H4a=0.;h5a=0.;h6a=0.
ht1a=0.;ht3a=0.;ht4a=0.;ht5a=0.;ht6a=0.
htr1a=0.;htr3a=0.;htr4a=0.;htr5a=0.;htr6a=0.

Ps1a=0.;Ps3a=0.;Ps4a=0.;Ps5a=0.;Ps6a=0.
Pt1a=0.;Pt3a=0.;Pt4a=0.;Pt5a=0.;Pt6a=0.
Ptr1a=0.;Ptr3a=0.;Ptr4a=0.;Ptr5a=0.;Ptr6a=0.

T1a=0.;T3a=0.;T4a=0.;T5a=0.;T6a=0.
Tt1a=0.;Tt3a=0.;Tt4a=0.;Tt5a=0.;Tt6a=0.

V1a=0.;V3a=0.;V4a=0.;V5a=0.;V6a=0.
W1a=0.;W3a=0.;W4a=0.;W5a=0.;W6a=0.

zsa=0.;zra=0.;etsa=0.;etta=0.
estagea=0.;mexta=0.;Pra=0.

!indir=trim(adjustl(indir))//'avg_values'
do i=1,n
write(*,'(A40,1x,I6)')'table:reading time= ',i
write(noint,'(I4)')i
filein=trim(adjustl(indir))//trim(adjustl(noint))//ext
open(unit=33,file=filein,status='unknown')
read(33,*)Pdatum !A1
read(33,*)ht1(i);ht1a=ht1a+ht1(i) !A2
read(33,*)ht3(i);ht3a=ht3a+ht3(i) !A3
read(33,*)ht4(i);ht4a=ht4a+ht4(i) !A4
read(33,*)ht5(i);ht5a=ht5a+ht5(i) !A5
read(33,*)ht6(i);ht6a=ht6a+ht6(i) !A6
read(33,*)htr1(i);htr1a=htr1a+htr1(i) !A7
read(33,*)htr3(i);htr3a=htr3a+htr3(i) !A8
read(33,*)htr4(i);htr4a=htr4a+htr4(i) !A9
read(33,*)htr5(i);htr5a=htr5a+htr5(i) !A10
read(33,*)htr6(i);htr6a=htr6a+htr6(i) !A11
read(33,*)Pratio(i);Pra=Pra+Pratio(i) !A12
read(33,*)Ps1(i);Ps1a=Ps1a+Ps1(i) !A13
read(33,*)Ps3(i);Ps3a=Ps3a+Ps3(i) !A14
read(33,*)Ps4(i);Ps4a=Ps4a+Ps4(i) !A15
read(33,*)Ps5(i);Ps5a=Ps5a+Ps5(i) !A16
read(33,*)Ps6(i);Ps6a=Ps6a+Ps6(i) !A17
read(33,*)Pt1(i);Pt1a=Pt1a+Pt1(i) !A18

```

```

read(33,*)Pt3(i);Pt3a=Pt3a+Pt3(i) !A19
read(33,*)Pt4(i);Pt4a=Pt4a+Pt4(i) !A20
read(33,*)Pt5(i);Pt5a=Pt5a+Pt5(i) !A21
read(33,*)Pt6(i);Pt6a=Pt6a+Pt6(i) !A22
read(33,*)Ptr1(i);Ptr1a=Ptr1a+Ptr1(i) !A23
read(33,*)Ptr3(i);Ptr3a=Ptr3a+Ptr3(i) !A24
read(33,*)Ptr4(i);Ptr4a=Ptr4a+Ptr4(i) !A25
read(33,*)Ptr5(i);Ptr5a=Ptr5a+Ptr5(i) !A26
read(33,*)Ptr6(i);Ptr6a=Ptr6a+Ptr6(i) !A27
read(33,*)T1(i);T1a=T1a+T1(i) !A28
read(33,*)T3(i);T3a=T3a+T3(i) !A29
read(33,*)T4(i);T4a=T4a+T4(i) !A30
read(33,*)T5(i);T5a=T5a+T5(i) !A31
read(33,*)T6(i);T6a=T6a+T6(i) !A32
read(33,*)Tt1(i);Tt1a=Tt1a+Tt1(i) !A33
read(33,*)Tt3(i);Tt3a=Tt3a+Tt3(i) !A34
read(33,*)Tt4(i);Tt4a=Tt4a+Tt4(i) !A35
read(33,*)Tt5(i);Tt5a=Tt5a+Tt5(i) !A36
read(33,*)Tt6(i);Tt6a=Tt6a+Tt6(i) !A37
read(33,*)V1(i) !A38
read(33,*)V3(i) !A39
read(33,*)V4(i) !A40
read(33,*)V5(i) !A41
read(33,*)V6(i) !A42
read(33,*)W1(i) !A43
read(33,*)W3(i) !A44
read(33,*)W4(i) !A45
read(33,*)W5(i) !A46
read(33,*)W6(i) !A47
read(33,*)h1(i) !A48
read(33,*)h3(i) !A49
read(33,*)h4(i) !A50
read(33,*)h5(i) !A51
read(33,*)h6(i) !A52
read(33,*)zs(i) !A53
read(33,*)zr(i) !A54
read(33,*)ets(i) !A55
read(33,*)ett(i) !A56
read(33,*)estage(i) !A57
read(33,*)mext(i);mext(i)=abs(mext(i)) !A58
close(33)
V1a=V1a+V1(i)
V3a=V3a+V3(i)
V4a=V4a+V4(i)
V5a=V5a+V5(i)
V6a=V6a+V6(i)
W1a=W1a+W1(i)
W3a=W3a+W3(i)
W4a=W4a+W4(i)
W5a=W5a+W5(i)
W6a=W6a+W6(i)
h1a=h1a+h1(i)
h3a=h3a+h3(i)
h4a=h4a+h4(i)
h5a=h5a+h5(i)
h6a=h6a+h6(i)
zsa=zsa+zs(i)
zra=zra+zr(i)
etsa=etsa+ets(i)
etta=etta+ett(i)
mexta=mexta+mext(i)
estagea=estagea+estage(i)
end do

!get averages in case

V1a=V1a/float(n)
V3a=V3a/float(n)

```

```

V4a=V4a/float(n)
V5a=V5a/float(n)
V6a=V6a/float(n)
W1a=W1a/float(n)
W3a=W3a/float(n)
W4a=W4a/float(n)
W5a=W5a/float(n)
W6a=W6a/float(n)
h1a=h1a/float(n)
h3a=h3a/float(n)
h4a=h4a/float(n)
h5a=h5a/float(n)
h6a=h6a/float(n)
ht1a=ht1a/float(n)
ht3a=ht3a/float(n)
ht4a=ht4a/float(n)
ht5a=ht5a/float(n)
ht6a=ht6a/float(n)
htr1a=htr1a/float(n)
htr3a=htr3a/float(n)
htr4a=htr4a/float(n)
htr5a=htr5a/float(n)
htr6a=htr6a/float(n)
T1a=T1a/float(n)
T3a=T3a/float(n)
T4a=T4a/float(n)
T5a=T5a/float(n)
T6a=T6a/float(n)
Tt1a=Tt1a/float(n)
Tt3a=Tt3a/float(n)
Tt4a=Tt4a/float(n)
Tt5a=Tt5a/float(n)
Tt6a=Tt6a/float(n)
Ps1a=Ps1a/float(n)
Ps3a=Ps3a/float(n)
Ps4a=Ps4a/float(n)
Ps5a=Ps5a/float(n)
Ps6a=Ps6a/float(n)
Pt1a=Pt1a/float(n)
Pt3a=Pt3a/float(n)
Pt4a=Pt4a/float(n)
Pt5a=Pt5a/float(n)
Pt6a=Pt6a/float(n)
Ptr1a=Ptr1a/float(n)
Ptr3a=Ptr3a/float(n)
Ptr4a=Ptr4a/float(n)
Ptr5a=Ptr5a/float(n)
Ptr6a=Ptr6a/float(n)
etta=etta/float(n)
etsa=etsa/float(n)
zra=zra/float(n)
zsa=zsa/float(n)
mexta=mexta/float(n)
estagea=estagea/float(n)
Pra=Pra/float(n)

```

```

open(unit=33,file='table_averages.dat',status='unknown')
write(33,7421)'Static Pressure= ',ps1a,ps3a,ps4a,ps5a,ps6a
write(33,7421)'Total Pressure= ',pt1a,pt3a,pt4a,pt5a,pt6a
write(33,7421)'Relative Total Pressure= ',ptr1a,ptr3a,ptr4a,ptr5a,ptr6a
write(33,7421)'Static Enthalpy= ',h1a,h3a,h4a,h5a,h6a
write(33,7421)'Total Enthalpy= ',ht1a,ht3a,ht4a,ht5a,ht6a
write(33,7421)'Relative Total Enthalpy= ',htr1a,htr3a,htr4a,htr5a,htr6a
write(33,7421)'Static Temperature= ',t1a,t3a,t4a,t5a,t6a
write(33,7421)'Total Temperature= ',tt1a,tt3a,tt4a,tt5a,tt6a
write(33,7422)'Stator Loss=',zsa

```

```

write(33,7422)'Rotor Loss=',zra
write(33,7422)'PR= ',zra
write(33,7422)'eta_ts= ',etsa
write(33,7422)'eta_tt= ',etta
write(33,7422)'massflow= ',mexta

7421 format(A30,1x,5(F12.3,1x))
7422 format(A30,1x,F12.3)

close(33)

open(unit=33,file='vtime.dat',status='unknown')
do i=1,n
write(33,7433),float(i)*dt,float(i)*dt/dpitch(1),pratio(i),mext(i),ets(i),ett(i),zs(i),zr(i)
end do
7433 format(8(F25.10,1x))
close(33)

deallocate(h1,h3,h4,h5,h6,stat=iq)
deallocate(ht1,ht3,ht4,ht5,ht6,stat=iq)
deallocate(htr1,htr3,htr4,htr5,htr6,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"
deallocate(ps1,ps3,ps4,ps5,ps6,stat=iq)
deallocate(pt1,pt3,pt4,pt5,pt6,stat=iq)
deallocate(ptr1,ptr3,ptr4,ptr5,ptr6,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"
deallocate(t1,t3,t4,t5,t6,stat=iq)
deallocate(tt1,tt3,tt4,tt5,tt6,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"
deallocate(V1,V3,V4,V5,V6,stat=iq)
deallocate(W1,W3,W4,W5,W6,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"
deallocate(zs,zr,ett,ets,estage,mext,pratio,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

end subroutine

subroutine tline_analysis(n,ns,t,norm)

use params
use averager

implicit none
integer::n,ns,t,norm,iq

real(8),allocatable,dimension(:,:,:)::span,pt,ps,ptr,mm,mmr,ts,tt,ttr,hs,ht,htr,s
real(8),allocatable,dimension(:,:,:)::v,w,vb2b,wb2b,rho,tcd,vc,wc,u,c,va,wa,vr,wr
real(8),allocatable,dimension(:,:,:)::vu,vv,vw,wu,ww,vm,wm
real(8),allocatable,dimension(:,:,:)::alpha,beta,gamma,gamma1,radius
!averaged values
!real(8),dimension(n,ns)::v_a,vaa,vca,vra,w_a,waa,wca,wra
!real(8),dimension(n,ns)::teda,mma,mmra,vb2ba,wb2ba,psa,pta,ptr
!real(8),dimension(n,ns)::hsa,hta,htra,tsa,tta,ttra
!real(8),dimension(n,ns)::alphaa,betaa,gammaa,gamma1a
!real(8),dimension(n,ns)::rhoa,sp,sa,ua,rad
real(8),allocatable,dimension(:,:,:)::v_a,vaa,vca,vra,w_a,waa,wca,wra
real(8),allocatable,dimension(:,:,:)::teda,mma,mmra,vb2ba,wb2ba,psa,pta,ptr
real(8),allocatable,dimension(:,:,:)::hsa,hta,htra,tsa,tta,ttra
real(8),allocatable,dimension(:,:,:)::alphaa,betaa,gammaa,gamma1a
real(8),allocatable,dimension(:,:,:)::rhoa,sp,sa,ua,rad,massflow
real(8),allocatable,dimension(:)::z_s,z_r,ss,sr,sst,eta_s,eta_r,eta_st,eta_sti,temp,temp1

real(8)::z_r_avg,z_s_avg,eta_s_avg,eta_r_avg,eta_st_avg,eta_sti_avg
real(8)::dhs1,dhs2,dhs3,tr
real(8)::vvec_i,num,den,ps3,ps4,ps5,t3,t4,t5,t4s,t5s,t5s1
real(8)::lm

```

```

real(8)::drks,drkt,drktr !these are for normalization (druck as in pressure)

real(8)::integ

character(len=80)::indir,outdir,time,infile,feff,outfile,dum_string,fout1,fout2,cmd

character(len=4)::ext,nis
character(len=1500)::superstring

tr=float(t)

allocate(span(n,ns,t),pt(n,ns,t),ps(n,ns,t),ptr(n,ns,t),stat=iq)
allocate(mm(n,ns,t),mmr(n,ns,t),ts(n,ns,t),tt(n,ns,t),ttr(n,ns,t),stat=iq)
allocate(hs(n,ns,t),ht(n,ns,t),htr(n,ns,t),s(n,ns,t),stat=iq)
allocate(v(n,ns,t),w(n,ns,t),vb2b(n,ns,t),wb2b(n,ns,t),rho(n,ns,t),stat=iq)
allocate(ted(n,ns,t),vc(n,ns,t),wc(n,ns,t),u(n,ns,t),c(n,ns,t),stat=iq)
allocate(va(n,ns,t),wa(n,ns,t),vr(n,ns,t),wr(n,ns,t),stat=iq)
allocate(vu(n,ns,t),vv(n,ns,t),vw(n,ns,t),wu(n,ns,t),wv(n,ns,t),ww(n,ns,t),stat=iq)
allocate(vm(n,ns,t),wm(n,ns,t),alpha(n,ns,t),beta(n,ns,t),gamma(n,ns,t),stat=iq)
allocate(gamma1(n,ns,t),radius(n,ns,t),stat=iq)
allocate(massflow(ns,t),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"

allocate(v_a(n,ns),vaa(n,ns),vca(n,ns),vra(n,ns),w_a(n,ns),waa(n,ns),wca(n,ns),wra(n,ns),stat=iq)
allocate(teda(n,ns),mma(n,ns),mmra(n,ns),vb2ba(n,ns),wb2ba(n,ns),psa(n,ns),pta(n,ns),ptr(n,ns),stat=iq)
allocate(hsa(n,ns),hta(n,ns),htra(n,ns),tsa(n,ns),tta(n,ns),ttra(n,ns),stat=iq)
allocate(alphaa(n,ns),betaa(n,ns),gammaa(n,ns),gamma1a(n,ns),stat=iq)
allocate(rhoa(n,ns),sp(n,ns),sa(n,ns),ua(n,ns),rad(n,ns),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"

allocate(z_s(n),z_r(n),ss(n),sr(n),sst(n),eta_s(n),eta_r(n),stat=iq)
allocate(eta_st(n),eta_sti(n),temp(n),temp1(n),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"

write(rpms,'(F8.0)')rpm
rpms=adjustl(rpms);rpms=rpms(1:4);
umean=omega*0.5*(rhub+rtip)
ext='.dat'
indir='Lineplots/conv/'
outdir='avg_out'

!clear system
dum_string='rm -r '//outdir
dum_string=trim(adjustl(dum_string))

call system(dum_string)

dum_string='mkdir '//outdir
dum_string=trim(adjustl(dum_string))
call system(dum_string)

do j=1,ns
write(nis,'(I1)')j+2
nis=adjustl(nis)
do k=1,t
write(*,'(A30,1x,I3,1x,A10,I6)')'line: reading station',j+2,'for time= ',k
write(time,'(I4)')k
time=adjustl(time)
infile=trim(indir)//'Line_Station'//trim(nis)//'_'//trim(time)//trim(ext)
open(unit=3,file=infile,status='unknown')
open(unit=4,file='test.dat',status='unknown')
do i=1,n
read(3,*)span(i,j,k),pt(i,j,k),ptr(i,j,k),ps(i,j,k),mm(i,j,k),&

```



```

mmr(i,j,k),ts(i,j,k),tt(i,j,k),ttr(i,j,k),rho(i,j,k),&
s(i,j,k),hs(i,j,k),ht(i,j,k),htr(i,j,k),u(i,j,k),v(i,j,k),&
vu(i,j,k),vv(i,j,k),vw(i,j,k),va(i,j,k),vc(i,j,k),vr(i,j,k),&
wm(i,j,k),w(i,j,k),wu(i,j,k),wv(i,j,k),ww(i,j,k),wa(i,j,k),&
wc(i,j,k),wr(i,j,k),wm(i,j,k),alpha(i,j,k),beta(i,j,k),gamma(i,j,k),&
vb2b(i,j,k),wb2b(i,j,k),ted(i,j,k)

vb2b(i,j,k)=sqrt(va(i,j,k)**2+vc(i,j,k)**2)
alpha(i,j,k)=acos(vc(i,j,k)/vb2b(i,j,k))
gamma(i,j,k)=atan(vr(i,j,k)/va(i,j,k))
gamma1(i,j,k)=asin(vr(i,j,k)/v(i,j,k))

radius(i,j,k)=rhub+span(i,j,k)*(rtip-rhub)
u(i,j,k)=omega*radius(i,j,k)

wc(i,j,k)=vc(i,j,k)-u(i,j,k)
wa(i,j,k)=va(i,j,k)
wr(i,j,k)=vr(i,j,k)
wb2b(i,j,k)=sqrt(wa(i,j,k)**2+wc(i,j,k)**2)
w(i,j,k)=sqrt(wb2b(i,j,k)**2+wr(i,j,k)**2)
vvec_i=vb2b(i,j,k)*cos(alpha(i,j,k))
beta(i,j,k)=acos(wc(i,j,k)/wb2b(i,j,k))

pt(i,j,k)=ps(i,j,k)+0.5*rho(i,j,k)*(v(i,j,k)*v(i,j,k))
ptr(i,j,k)=ps(i,j,k)+0.5*rho(i,j,k)*(w(i,j,k)*w(i,j,k)-u(i,j,k)*u(i,j,k))

ht(i,j,k)=hs(i,j,k)+0.5*v(i,j,k)*v(i,j,k)
htr(i,j,k)=hs(i,j,k)+0.5*(w(i,j,k)*w(i,j,k)-u(i,j,k)*u(i,j,k))

c(i,j,k)=v(i,j,k)/mmr(i,j,k)
mmr(i,j,k)=w(i,j,k)/c(i,j,k)
end do
close(3)
close(4)
temp1(1:n)=radius(1:n,j,k)*rho(1:n,j,k)*va(1:n,j,k)
den=integ(n,radius(1:n,j,k),temp1(1:n))
massflow(j,k)=den*2*pi
end do
end do
!convert pressure to kPa
!correct pressure
ps=ps+pcor;pt=pt+pcor;ptr=ptr+pcor
ps=ps/1000.0;pt=pt/1000.0;ptr=ptr/1000.0
!convert entropy and enthalpy to kJ/kg/K or kJ/kg
s=s/1000.0;hs=hs/1000.0;ht=ht/1000.0;htr=htr/1000.0
open(unit=111,file='massflow.out',status='unknown')

do i=1,t
write(111,112)float(i)*dt,massflow(ns,i)
end do
112 format(F8.5,1x,F8.5)
close(111)

do j=1,ns
do i=1,n
rad(i,j)=0;psa(i,j)=0;pta(i,j)=0;ptra(i,j)=0
v_a(i,j)=0;vaa(i,j)=0;vra(i,j)=0;vca(i,j)=0
w_a(i,j)=0;waa(i,j)=0;wra(i,j)=0;wca(i,j)=0
tsa(i,j)=0;tta(i,j)=0;ttra(i,j)=0
hsa(i,j)=0;hta(i,j)=0;htra(i,j)=0
alphaa(i,j)=0;betaa(i,j)=0;gammaa(i,j)=0;gamma1a(i,j)=0
sa(i,j)=0;rhoa(i,j)=0;mma(i,j)=0;mmra(i,j)=0;
ua(i,j)=0;vb2ba(i,j)=0;wb2ba(i,j)=0;rad(i,j)=0

do k=1,t
psa(i,j)=psa(i,j)+ps(i,j,k)
pta(i,j)=pta(i,j)+pt(i,j,k)

```

```

ptr(i,j)=ptr(i,j)+ptr(i,j,k)
v_a(i,j)=v_a(i,j)+v(i,j,k)
vaa(i,j)=vaa(i,j)+va(i,j,k)
vra(i,j)=vra(i,j)+vr(i,j,k)
vca(i,j)=vca(i,j)+vc(i,j,k)
w_a(i,j)=w_a(i,j)+w(i,j,k)
waa(i,j)=waa(i,j)+wa(i,j,k)
wra(i,j)=wra(i,j)+wr(i,j,k)
wca(i,j)=wca(i,j)+wc(i,j,k)
tsa(i,j)=tsa(i,j)+ts(i,j,k)
tta(i,j)=tta(i,j)+tt(i,j,k)
ttra(i,j)=ttra(i,j)+ttr(i,j,k)
hsa(i,j)=hsa(i,j)+hs(i,j,k)
hta(i,j)=hta(i,j)+ht(i,j,k)
htra(i,j)=htra(i,j)+htr(i,j,k)
alphaa(i,j)=alphaa(i,j)+alpha(i,j,k)
betaa(i,j)=betaa(i,j)+beta(i,j,k)
gammaa(i,j)=gammaa(i,j)+gamma(i,j,k)
gamma1a(i,j)=gamma1a(i,j)+gamma1(i,j,k)
sa(i,j)=sa(i,j)+s(i,j,k)
rhoa(i,j)=rhoa(i,j)+rho(i,j,k)
mma(i,j)=mma(i,j)+mm(i,j,k)
mmra(i,j)=mmra(i,j)+mmr(i,j,k)
teda(i,j)=teda(i,j)+ted(i,j,k)
vb2ba(i,j)=vb2ba(i,j)+vb2b(i,j,k)
wb2ba(i,j)=wb2ba(i,j)+wb2b(i,j,k)
rad(i,j)=rad(i,j)+radius(i,j,k)
sp(i,j)=sp(i,j)+span(i,j,k)

end do

psa(i,j)=psa(i,j)/tr
pta(i,j)=pta(i,j)/tr
ptr(i,j)=ptr(i,j)/tr
v_a(i,j)=v_a(i,j)/tr
vaa(i,j)=vaa(i,j)/tr
vra(i,j)=vra(i,j)/tr
vca(i,j)=vca(i,j)/tr
w_a(i,j)=w_a(i,j)/tr
waa(i,j)=waa(i,j)/tr
wra(i,j)=wra(i,j)/tr
wca(i,j)=wca(i,j)/tr
tsa(i,j)=tsa(i,j)/tr
tta(i,j)=tta(i,j)/tr
ttra(i,j)=ttra(i,j)/tr
hsa(i,j)=hsa(i,j)/tr
hta(i,j)=hta(i,j)/tr
htra(i,j)=htra(i,j)/tr
mma(i,j)=mma(i,j)/tr
mmra(i,j)=mmra(i,j)/tr
rhoa(i,j)=rhoa(i,j)/tr
rad(i,j)=rad(i,j)/tr
sp(i,j)=sp(i,j)/tr
alphaa(i,j)=alphaa(i,j)/tr
betaa(i,j)=betaa(i,j)/tr
gammaa(i,j)=gammaa(i,j)/tr
gamma1a(i,j)=gamma1a(i,j)/tr
ua(i,j)=rad(i,j)*omega
end do
end do

!get lineplot mass average of temporal average
do j=1,ns
temp1(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)
den=integ(n,rad(1:n,j),temp1(1:n))

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*psa(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))

```

```

psavg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*pta(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
ptavg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*ptra(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
ptravg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*tsa(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
tsavg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*tta(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
ttavg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*ttra(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
ttravg(j)=num/den
mdot(j)=den*2*pi

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*hsa(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
hsavg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*hta(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
htavg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*htra(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
htravg(j)=num/den

temp(1:n)=rad(1:n,j)*rhoa(1:n,j)*vaa(1:n,j)*vca(1:n,j)
num=integ(n,rad(1:n,j),temp(1:n))
vcavg(j)=num/den

end do
ps3=psavg(1);ps4=psavg(2);ps5=psavg(3);t3=tsavg(1);t4=tsavg(2);t5=tsavg(3)

do i=1,ns
psmax(i)=maxval(psa(1:n,i))
ptmax(i)=maxval(pta(1:n,i))
ptrmax(i)=maxval(ptra(1:n,i))
psmin(i)=minval(psa(1:n,i))
ptmin(i)=minval(pta(1:n,i))
ptrmin(i)=minval(ptra(1:n,i))

end do

t4s=t3*(ps4/ps3)**pwr
t5s=t4*(ps5/ps4)**pwr
t5s1=t3*(ps5/ps3)**pwr

dhs1=cp*(t3-t4s)
dhs2=cp*(t4-t5s)
dhs3=cp*(t3-t5s1)

do i=1,n
ss(i)=0.5*(sp(i,1)+sp(i,2))
sr(i)=0.5*(sp(i,2)+sp(i,3))
sst(i)=0.5*(sp(i,1)+sp(i,3))

eta_s(i)=(0.5*v_a(i,2)**2)/(dhs1+0.5*v_a(i,1)**2)
eta_r(i)=(0.5*(w_a(i,3)**2-u(i,3,1)**2))/(dhs2+0.5*(w_a(i,2)**2-u(i,2,1)**2))
eta_st(i)=(0.5*(w_a(i,3)**2-u(i,3,1)**2+v_a(i,2)**2))/&

```

```

(dhs3+0.5*(v_a(i,1)**2+w_a(i,2)**2-u(i,2,1)**2))
eta_sti(i)=(1-(tta(i,3)/tta(i,1)))/(1-(psa(i,3)/pta(i,1))*pwr)

z_s(i)=(pta(i,1)-pta(i,2))/(ptavg(1)-psavg(2))
z_r(i)=(ptrav(2)-ptrav(3))/(ptravg(2)-psavg(3))
end do
z_s_avg=integ(n,ss(1:n),z_s(1:n))
z_r_avg=integ(n,sr(1:n),z_r(1:n))
eta_s_avg=integ(n,sr(1:n),eta_s(1:n))
eta_r_avg=integ(n,sr(1:n),eta_r(1:n))
eta_st_avg=integ(n,sr(1:n),eta_st(1:n))
eta_sti_avg=integ(n,sr(1:n),eta_sti(1:n))
write(*,*)z_s_avg,z_r_avg,eta_s_avg,eta_r_avg,eta_st_avg

PR=psavg(1)/psavg(ns)
co=sqrt(2*(htavg(1)-htavg(ns))/float(nstages))
u_co=umean/co
feff=TRIM(outdir)//TRIM('/Eff_Summary_')//TRIM(rpms)//TRIM('.dat')
feff=TRIM(ADJUSTL(feff))
OPEN(UNIT=214,FILE=feff,STATUS='UNKNOWN')
write(214,'(A30,1x,F10.5)')RPM= ',RPM
write(214,'(A30,1x,F10.5)')Pressure Ratio= ',PR
write(214,'(A30,1x,F10.5)')MassFlow',mdot(ns)
!write(214,'(A30,1x,F10.5)')U/Co= ',U_CO
!write(214,'(A30,1x,F10.5)')Machine Efficiency t-s= ',ETA_TS
!write(214,'(A30,1x,F10.5)')Machine Efficiency t-t= ',ETA_TT
write(214,'(A30,1x,F10.5)')Stator 2 Efficiency t-s= ',ETA_S_AVG
write(214,'(A30,1x,F10.5)')Rotor 2 Efficiency= ',ETA_R_AVG
write(214,'(A30,1x,F10.5)')Stage 2 Efficiency= ',ETA_ST_AVG
write(214,'(A30,1x,F10.5)')Stator 2 Loss Coefficient= ',Z_S_AVG
write(214,'(A30,1x,F10.5)')Rotor 2 Loss Coefficient= ',Z_R_AVG
!write(214,'(A30,1x,F10.5)')Rotor 1 Reaction= ',(hsavg(2)-hsavg(3))/(hsavg(1)-hsavg(3))
write(214,'(A30,1x,F10.5)')Rotor 2 Reaction= ',(hsavg(2)-hsavg(3))/(hsavg(1)-hsavg(3))
write(214,'(A30,1x,F10.5)')Stage 1 Flow Coefficient= ',vmavg(1)/umean
write(214,'(A30,1x,F10.5)')Stage 2 Flow Coefficient= ',vmavg(3)/umean
!lm=Umean*VCavg(2)+Umean*VCavg(3)
!write(214,'(A30,1x,F10.5)')Stage 1 Load Coefficient= ',lm/(umean*umean)
!lm=Umean*VCavg(4)+Umean*VCavg(5)
!write(214,'(A30,1x,F10.5)')Stage 2 Load Coefficient= ',lm/(umean*umean)

!WRITE(214,'(8(F10.5,1x))')RPM,PR,U_CO,ETA_TS,ETA_S_AVG,ETA_R_AVG&
!      ,ETA_ST_AVG,Z_S_AVG,Z_R_AVG
CLOSE(214)

FEFF=TRIM(outdir)//'/Eff_Dump_'//TRIM(RPMS)//'.dat'
OPEN(UNIT=215,FILE=FEFF,Status='unknown')
WRITE(215,1002)'Rotation Speed [1/min]= ',RPM
WRITE(215,1002)'Avg Stator Total Pressure Loss Coefficient= ',&
Z_S_AVG
WRITE(215,1002)'Avg Rotor Total Pressure Loss Coefficient= ',&
Z_R_AVG
WRITE(215,1002)'Avg Stator Efficiency= ',ETA_S_AVG
WRITE(215,1002)'Avg Rotor Efficiency= ',ETA_R_AVG
WRITE(215,1002)'Avg Stage(2) Efficiency= ',ETA_ST_AVG
WRITE(215,1002)'Avg Stage Isentropic Eff= ',ETA_STI_AVG
WRITE(215,1002)'U_mean = ',UMEAN
!WRITE(215,1002)'Co = ',CO
!WRITE(215,1002)'U/Co= ',U_CO

CLOSE(215)

1002 FORMAT(A45,1x,F10.5)

feff=trim('avg_Loss_')//trim(RPMS)//trim('.dat')
feff=trim(outdir)//'/trim(adjustl(feff))

open(unit=212,file=feff,status='unknown')

```

```

write(212,*)'TITLE="Averaged Loss Summary"'
write(212,'(A120)')'VARIABLES="Span Stator","Stator Total Pressure Loss Coefficient",&
//'"Span Rotor","Rotor Total Pressure Loss Coefficient"'

feff=trim('avg_Eff_')//trim(RPMS)//trim('.dat')
feff=trim(outdir)///'//trim(adjust1(feff))
open(unit=213,file=feff,status='unknown')
write(213,*)'TITLE="Averaged Efficiency Summary"'
write(213,'(A115)')'VARIABLES="Span Stator","Stator Efficiency",&
//'"Span Rotor","Rotor Efficiency",&
//'"Span Stage","Stage(2) Efficiency"'

do i=1,n
  write(212,1000)ss(i),z_s(i),sr(i),z_r(i)
  write(213,1001)ss(i),eta_s(i),sr(i),eta_r(i),sst(i),eta_st(i)
end do
close(212);close(213)
1000 format(4(F8.5,1x))
1001 format(6(F8.5,1x))

fout1=trim(outdir)///'//trim('tcfdplot_')//trim(rpms)//trim('.dat')
fout1=trim(adjust1(fout1))
open(unit=14,file=fout1,status='unknown')
call get_supstring(ver,norm,supstring)
write(14,'(A1500)')supstring
do j=1,ns
  if(j == 1)then
    write(14,*)'ZONE T="Station 3 Averaged CFD-URANS"'
  elseif(j==2)then
    write(14,*)'ZONE T="Station 4 Averaged CFD-URANS"'
  elseif(j==3)then
    write(14,*)'ZONE T="Station 5 Averaged CFD-URANS"'
  end if
  do i=1,n
    if((norm<=0).or.(norm>3))then
      write(14,1978)sp(i,j),pta(i,j),ptr(i,j),psa(i,j),v_a(i,j),vca(i,j),&
        vra(i,j),vaa(i,j),w_a(i,j),wca(i,j),wra(i,j),waa(i,j),ua(i,j),&
        gammaa(i,j)*180.0/pi,alphaa(i,j)*180./pi,betaa(i,j)*180./pi,&
        mma(i,j),mmra(i,j),tsa(i,j),tta(i,j),ttra(i,j),cp,sa(i,j),hsa(i,j),&
        hta(i,j),htra(i,j),rhoa(i,j),cp,cp,gamma1a(i,j),cp
    elseif(norm==1)then
      drks=(psa(i,j)-psmin(j))/(psmax(j)-psmin(j))
      drkt=(pta(i,j)-ptmin(j))/(ptmax(j)-ptmin(j))
      drktr=(ptr(i,j)-ptrmin(j))/(ptrmax(j)-ptrmin(j))
      write(14,1978)sp(i,j),drkt,drktr,drks,v_a(i,j),vca(i,j),&
        vra(i,j),vaa(i,j),w_a(i,j),wca(i,j),wra(i,j),waa(i,j),ua(i,j),&
        gammaa(i,j)*180.0/pi,alphaa(i,j)*180./pi,betaa(i,j)*180./pi,&
        mma(i,j),mmra(i,j),tsa(i,j),tta(i,j),ttra(i,j),cp,sa(i,j),hsa(i,j),&
        hta(i,j),htra(i,j),rhoa(i,j),cp,cp,gamma1a(i,j),cp
    elseif(norm==2)then
      drks=psa(i,j)/psavg(1)
      drkt=pta(i,j)/ptavg(1)
      drktr=ptr(i,j)/ptravg(1)
      write(14,1978)sp(i,j),drkt,drktr,drks,v_a(i,j),vca(i,j),&
        vra(i,j),vaa(i,j),w_a(i,j),wca(i,j),wra(i,j),waa(i,j),ua(i,j),&
        gammaa(i,j)*180.0/pi,alphaa(i,j)*180./pi,betaa(i,j)*180./pi,&
        mma(i,j),mmra(i,j),tsa(i,j),tta(i,j),ttra(i,j),cp,sa(i,j),hsa(i,j),&
        hta(i,j),htra(i,j),rhoa(i,j),cp,cp,gamma1a(i,j),cp
    elseif(norm==3)then
      drks=psa(i,j)/psmax(1)
      drkt=pta(i,j)/ptmax(1)
      drktr=ptr(i,j)/ptrmax(1)
      write(14,1978)sp(i,j),drkt,drktr,drks,v_a(i,j),vca(i,j),&
        vra(i,j),vaa(i,j),w_a(i,j),wca(i,j),wra(i,j),waa(i,j),ua(i,j),&
        gammaa(i,j)*180.0/pi,alphaa(i,j)*180./pi,betaa(i,j)*180./pi,&
        mma(i,j),mmra(i,j),tsa(i,j),tta(i,j),ttra(i,j),cp,sa(i,j),hsa(i,j),&
        hta(i,j),htra(i,j),rhoa(i,j),cp,cp,gamma1a(i,j),cp

```

```

endif
end do
end do
close(14)
1978 format(31(F15.5,1x))

!gnuplot output
outdir='mkdir line_gnu'
call system(outdir)

outdir='line_gnu/'
outfile='Ps'

outfile=trim(adjustl(outdir))/trim(adjustl(outfile))

call make_gnu_line(n,ns,outfile,sp,psa)
outfile=trim(adjustl(outdir))//Pt';call make_gnu_line(n,ns,outfile,sp,pta)
outfile=trim(adjustl(outdir))//Ptr';call make_gnu_line(n,ns,outfile,sp,ptr)
outfile=trim(adjustl(outdir))//Hs';call make_gnu_line(n,ns,outfile,sp,hsa)
outfile=trim(adjustl(outdir))//Ht';call make_gnu_line(n,ns,outfile,sp,hta)
outfile=trim(adjustl(outdir))//Htr';call make_gnu_line(n,ns,outfile,sp,htra)
outfile=trim(adjustl(outdir))//S';call make_gnu_line(n,ns,outfile,sp,sa)
outfile=trim(adjustl(outdir))//Ts';call make_gnu_line(n,ns,outfile,sp,t)
outfile=trim(adjustl(outdir))//Tt';call make_gnu_line(n,ns,outfile,sp,tt)
outfile=trim(adjustl(outdir))//Ttr';call make_gnu_line(n,ns,outfile,sp,ttra)
outfile=trim(adjustl(outdir))//V';call make_gnu_line(n,ns,outfile,sp,v)
outfile=trim(adjustl(outdir))//W';call make_gnu_line(n,ns,outfile,sp,w)
outfile=trim(adjustl(outdir))//MM';call make_gnu_line(n,ns,outfile,sp,mm)
outfile=trim(adjustl(outdir))//MMR';call make_gnu_line(n,ns,outfile,sp,mmr)
outfile=trim(adjustl(outdir))//Alpha';call make_gnu_line(n,ns,outfile,sp,alphaa*r2d)
outfile=trim(adjustl(outdir))//Beta';call make_gnu_line(n,ns,outfile,sp,betaa*r2d)
outfile=trim(adjustl(outdir))//Gamma';call make_gnu_line(n,ns,outfile,sp,gammaa*r2d)
outfile=trim(adjustl(outdir))//Rho';call make_gnu_line(n,ns,outfile,sp,rhoa)

write(*,*)'line analysis complete'

! end do
!end do

deallocate(span,pt,ps,ptr,stat=iq)
deallocate(mm,mmr,ts,tt,ttr,stat=iq)
deallocate(hs,ht,htr,s,stat=iq)
deallocate(v,w,vb2b,wb2b,rho,stat=iq)
deallocate(ted,vc,wc,u,c,stat=iq)
deallocate(va,wa,vr,wr,stat=iq)
deallocate(vu,vv,vw,wu,ww,stat=iq)
deallocate(vm,wm,alpha,beta,gamma,stat=iq)
deallocate(gamma1,radius,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

deallocate(v_a,vaa,vca,vra,w_a,waa,wca,wra,stat=iq)
deallocate(teda,mma,mmra,vb2ba,wb2ba,psa,pta,ptr,stat=iq)
deallocate(hsa,hta,htra,tsa,tta,ttra,stat=iq)
deallocate(alphaa,betaa,gammaa,gamma1a,stat=iq)
deallocate(rhoa,sp,sa,ua,rad,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

!deallocate(psave,ptavg,ptravg,tsavg,ttavg,ttravg,stat=iq)
!deallocate(psmn,ptmin,ptrmin,psmax,ptmax,ptrmax,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

deallocate(z_s,z_r,ss,sr,sst,eta_s,eta_r,stat=iq)

```

```

deallocate(eta_st,eta_sti,temp,temp1,stat=iq)
deallocate(massflow,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

end subroutine

subroutine get_supstring(ver,norm,string)
integer::norm,ver
character(len=1500)::string
!*****
!norm=0 no normalization
!norm=1 in station max and min values
!norm=2 Station 3 average values
!norm=3 Station 3 maximum values
!ver =0 signifies newer version of tecplot format
!ver !=0 signifies older version of tecplot format
!*****

if (ver==0)then
if (norm==0)then
string='VARIABLES="Normalized Span r<sup>*</sup>",&
//'"Total Pressure [kPa]","Relative Total Pressure [kPa]",&
//'"Static Pressure [kPa]","Absolute Velocity [m/sec]",&
//'"Absolute Circumferential Velocity [m/sec]",&
//'"Absolute Radial Velocity [m/sec]",&
//'"Absolute Axial Velocity [m/sec]",&
//'"Relative Velocity [m/sec]",&
//'"Relative Circumferential Velocity [m/sec]",&
//'"Relative Radial Velocity [m/sec]",&
//'"Relative Axial Velocity [m/sec]",&
//'"Rotational Velocity [m/sec]",&
//'"<greek>g</greek> compound [deg]","<greek>a</greek> [deg]","<greek>b</greek> [deg]",&
//'"Mach Number","Relative Mach Number",&
//'"Static Temperature [K]",&
//'"Total Temperature [K]",&
//'"Relative Total Temperature [K]",&
//'"Specific Heat Capacity-C<sub>p</sub> [kJ/kg/K]",&
//'"Entropy [kJ/kg/K]","Enthalpy [kJ/kg]",&
//'"Total Enthalpy [kJ/kg]",&
//'"Relative Total Enthalpy [kJ/kg]",&
//'"Density-<greek>r</greek> [kg/m<sup>3</sup>]",&
//'"Dynamic Viscosity-<greek>m</greek> [kg/sec/m]",&
//'"Gas Constant-R [J/kg/K]",&
//'"<greek>g</greek> [deg]",&
//'"TKE [m<sup>2</sup>/sec<sup>2</sup>]",&
else
string='VARIABLES="Normalized Span r<sup>*</sup>",&
//'"Normalized Total Pressure","Normalized Relative Total Pressure",&
//'"Normalized Static Pressure","Absolute Velocity [m/sec]",&
//'"Absolute Circumferential Velocity [m/sec]",&
//'"Absolute Radial Velocity [m/sec]",&
//'"Absolute Axial Velocity [m/sec]",&
//'"Relative Velocity [m/sec]",&
//'"Relative Circumferential Velocity [m/sec]",&
//'"Relative Radial Velocity [m/sec]",&
//'"Relative Axial Velocity [m/sec]",&
//'"Rotational Velocity [m/sec]",&
//'"<greek>g</greek> compound [deg]","<greek>a</greek> [deg]","<greek>b</greek> [deg]",&
//'"Mach Number","Relative Mach Number",&
//'"Static Temperature [K]",&
//'"Total Temperature [K]",&
//'"Relative Total Temperature [K]",&
//'"Specific Heat Capacity-C<sub>p</sub> [J/kg/K]",&
//'"Entropy [kJ/kg/K]","Enthalpy [kJ/kg]",&
//'"Total Enthalpy [kJ/kg]",&
//'"Relative Total Enthalpy [kJ/kg]",&
//'"Density-<greek>r</greek> [kg/m<sup>3</sup>]",&
//'"Dynamic Viscosity-<greek>m</greek> (kg/sec/m)",&
//'"Gas Constant-R [J/kg/K]",&

```

```

        //'<greek>g</greek> [deg]', '&
        //'TKE [m<sup>2</sup>/sec<sup>2</sup>]'

end if

elseif(ver==1)then
if(norm==0)then
string='VARIABLES="Normalized Span r",'&
        //'Total Pressure [kPa],"Relative Total Pressure [kPa]", '&
        //'Static Pressure [kPa],"Absolute Velocity [m/sec]', '&
        //'Absolute Circumferential Velocity [m/sec]', '&
        //'Absolute Radial Velocity [m/sec]', '&
        //'Absolute Axial Velocity [m/sec]', '&
        //'Relative Velocity [m/sec]', '&
        //'Relative Circumferential Velocity [m/sec]', '&
        //'Relative Radial Velocity [m/sec]', '&
        //'Relative Axial Velocity [m/sec]', '&
        //'Rotational Velocity [m/sec]', '&
        //'g compound [deg]', 'a [deg]', 'b [deg]', '&
        //'Mach Number", "Relative Mach Number", '&
        //'Static Temperature [K]', '&
        //'Total Temperature [K]', '&
        //'Relative Total Temperature [K]', '&
        //'Specific Heat Capacity-Cp [J/kg/K]', '&
        //'Entropy [kJ/kg/K]", "Enthalpy [kJ/kg]', '&
        //'Total Enthalpy [kJ/kg]', '&
        //'Relative Total Enthalpy [kJ/kg]', '&
        //'Density-<greek>r</greek> [kg/m^3]', '      &
        //'Dynamic Viscosity-'m [kg/sec/m]', '&
        //'Gas Constant-R [J/kg/K]', '&
        //'g [deg]', '&
        //'TKE [m^2/sec^2]'

else
string='VARIABLES="Normalized Span r",'&
        //'Normalized Total Pressure", "Normalized Relative Total Pressure", '&
        //'Normalized Static Pressure", "Absolute Velocity [m/sec]', '&
        //'Absolute Circumferential Velocity [m/sec]', '&
        //'Absolute Radial Velocity [m/sec]', '&
        //'Absolute Axial Velocity [m/sec]', '&
        //'Relative Velocity [m/sec]', '&
        //'Relative Circumferential Velocity [m/sec]', '&
        //'Relative Radial Velocity [m/sec]', '&
        //'Relative Axial Velocity [m/sec]', '&
        //'Rotational Velocity [m/sec]', '&
        //'g compound [deg]', 'a [deg]', 'b [deg]', '&
        //'Mach Number", "Relative Mach Number", '&
        //'Static Temperature [K]', '&
        //'Total Temperature [K]', '&
        //'Relative Total Temperature [K]', '&
        //'Specific Heat Capacity-Cp [J/kg/K]', '&
        //'Entropy [kJ/kg/K]", "Enthalpy [kJ/kg]', '&
        //'Total Enthalpy [kJ/kg]', '&
        //'Relative Total Enthalpy [kJ/kg]', '&
        //'Density-<greek>r</greek> [kg/m^3]', '      &
        //'Dynamic Viscosity-'m [kg/sec/m]', '&
        //'Gas Constant-R [J/kg/K]', '&
        //'g [deg]', '&
        //'TKE [m^2/sec^2]'

endif

else
write(*,*)'header error'
string=' '
return

```



```

end if

end subroutine

subroutine get_supstringc(ver,norm,string)
implicit none
integer::norm,ver
character(len=1500)::string
!*****
!norm=0 no normalization
!norm=1 in station max and min values
!norm=2 Station 3 average values
!norm=3 Station 3 maximum values
!ver =0 signifies newer version of tecplot format
!ver !=0 signifies older version of tecplot format
!*****

if (ver==0)then
if (norm==0)then
string='VARIABLES="Y","Z",'&
//'"Total Pressure [kPa]"',&
//'"Relative Total Pressure [kPa]"',&
//'"Static Pressure [kPa]"',&
//'"Absolute Velocity [m/sec]"',&
//'"Absolute Circumferential Velocity [m/sec]"',&
//'"Absolute Radial Velocity [m/sec]"',&
//'"Absolute Axial Velocity [m/sec]"',&
//'"Relative Velocity [m/sec]"',&
//'"Relative Circumferential Velocity [m/sec]"',&
//'"Relative Radial Velocity [m/sec]"',&
//'"Relative Axial Velocity [m/sec]"',&
//'"Rotational Velocity [m/sec]"',&
//'"<greek>g</greek> compound [deg]"',<greek>a</greek> [deg]"',<greek>b</greek> [deg]"',&
//'"Mach Number","Relative Mach Number",&
//'"Static Temperature [K]"',&
//'"Total Temperature [K]"',&
//'"Relative Total Temperature [K]"',&
//'"Specific Heat Capacity - C<sub>p</sub> [J/kg/K]"',&
//'"Entropy [kJ/kg/K]"',&
//'"Enthalpy [kJ/kg]"',&
//'"Total Enthalpy [kJ/kg]"',&
//'"Relative Total Enthalpy [kJ/kg]"',&
//'"Density - <greek>r</greek> [kg/m<sup>3</sup>]"',&
//'"Dynamic Viscosity-<greek>m</greek> [kg/s/m]"',&
//'"Gas Constant - R [J/kg/K]"',&
//'"<greek>g</greek> [deg]"',&
//'"Turbulence Eddy Dissipation [1/s]"',&
//'"Turbulence Kinetic Energy [m<sup>2</sup>/sec<sup>2</sup>]"',&
//'"Turbulence Eddy Frequency [1/s]"',&
//'"Eddy Viscosity [kg m/s]"',&
//'"Rothalpy[kJ/Kg]"',&"angle"

else
string='VARIABLES="Y","Z",'&
//'"Normalized Total Pressure",&
//'"Normalized Relative Total Pressure",&
//'"Normalized Static Pressure",&
//'"Absolute Velocity [m/sec]"',&
//'"Absolute Circumferential Velocity [m/sec]"',&
//'"Absolute Radial Velocity [m/sec]"',&
//'"Absolute Axial Velocity [m/sec]"',&
//'"Relative Velocity [m/sec]"',&
//'"Relative Circumferential Velocity [m/sec]"',&
//'"Relative Radial Velocity [m/sec]"',&
//'"Relative Axial Velocity [m/sec]"',&
//'"Rotational Velocity [m/sec]"',&
//'"<greek>g</greek> compound [deg]"',<greek>a</greek> [deg]"',<greek>b</greek> [deg]"',&
//'"Mach Number","Relative Mach Number",&
//'"Static Temperature [K]"',&

```

```

//'"Total Temperature [K]",&
//'"Relative Total Temperature [K]",&
//'"Specific Heat Capacity - C<sub>p</sub> [J/kg/K]",&
//'"Entropy [kJ/kg/K]","Enthalpy [kJ/kg]",&
//'"Total Enthalpy [kJ/kg]",&
//'"Relative Total Enthalpy [kJ/kg]",&
//'"Density - <greek>r</greek> [kg/m<sup>3</sup>]",&
//'"Dynamic Viscosity-<greek>m</greek> [kg/s/m]",&
//'"Gas Constant - R [J/kg/K]",&
//'"<greek>g</greek> [deg]",&
//'"Turbulence Eddy Dissipation [1/s]",&
//'"Turbulence Kinetic Energy [m<sup>2</sup>/sec<sup>2</sup>]",&
//'"Turbulence Eddy Frequency [1/s]",&
//'"Eddy Viscosity [kg m/s]",&
//'"Rothalpy[kJ/Kg]","angle"

end if

elseif(ver==1)then
if(norm==0)then
string='VARIABLES="Y","Z",&
//'"Total Pressure [kPa]",&
//'"Relative Total Pressure [kPa]",&
//'"Static Pressure [kPa]",&
//'"Absolute Velocity [m/sec]",&
//'"Absolute Circumferential Velocity [m/sec]",&
//'"Absolute Radial Velocity [m/sec]",&
//'"Absolute Axial Velocity [m/sec]",&
//'"Relative Velocity [m/sec]",&
//'"Relative Circumferential Velocity [m/sec]",&
//'"Relative Radial Velocity [m/sec]",&
//'"Relative Axial Velocity [m/sec]",&
//'"Rotational Velocity [m/sec]",&
//'"'g compound [deg]","'a [deg]","'b [deg]",&
//'"Mach Number","Relative Mach Number",&
//'"Static Temperature [K]",&
//'"Total Temperature [K]",&
//'"Relative Total Temperature [K]",&
//'"Specific Heat Capacity - Cp [J/kg/K]",&
//'"Entropy [kJ/kg/K]","Enthalpy [kJ/kg]",&
//'"Total Enthalpy [kJ/kg]",&
//'"Relative Total Enthalpy [kJ/kg]",&
//'"Density - 'r [kg/m<sup>3</sup>]",&
//'"Dynamic Viscosity-'m [kg/s/m]",&
//'"Gas Constant-R [J/kg/K]",&
//'"'g [deg]",&
//'"Turbulence Eddy Dissipation [1/s]",&
//'"Turbulence Kinetic Energy [m^2/sec^2]",&
//'"Turbulence Eddy Frequency [1/s]",&
//'"Eddy Viscosity [kg m/s]",&
//'"Rothalpy[kJ/Kg]","angle"

else
string='VARIABLES="Y","Z",&
//'"Normalized Total Pressure",&
//'"Normalized Relative Total Pressure",&
//'"Normalized Static Pressure",&
//'"Absolute Velocity [m/sec]",&
//'"Absolute Circumferential Velocity [m/sec]",&
//'"Absolute Radial Velocity [m/sec]",&
//'"Absolute Axial Velocity [m/sec]",&
//'"Relative Velocity [m/sec]",&
//'"Relative Circumferential Velocity [m/sec]",&
//'"Relative Radial Velocity [m/sec]",&
//'"Relative Axial Velocity [m/sec]",&
//'"Rotational Velocity [m/sec]",&
//'"'g compound [deg]","'a [deg]","'b [deg]",&

```

```

        //'Mach Number","Relative Mach Number",&
        //'Static Temperature [K]",&
        //'Total Temperature [K]",&
        //'Relative Total Temperature [K]",&
        //'Specific Heat Capacity - Cp [J/kg/K]",&
        //'Entropy [kJ/kg/K]","Enthalpy [kJ/kg]",&
        //'Total Enthalpy [kJ/kg]",&
        //'Relative Total Enthalpy [kJ/kg]",&
        //'Density - 'r [kg/m<sup>3</sup>]",&
        //'Dynamic Viscosity-'m [kg/s/m]",&
        //'Gas Constant-R [J/kg/K]",&
        //'g [deg]",&
        //'Turbulence Eddy Dissipation [1/s]",&
        //'Turbulence Kinetic Energy [m^2/sec^2]",&
        //'Turbulence Eddy Frequency [1/s]",&
        //'Eddy Viscosity [kg m/s]",&
        //'Rothalpy[kJ/Kg]","angle"

endif

else
write(*,*)'header error'
string=' '
return
end if
end subroutine

subroutine contour_analysis(n,nx,ns,t,norm)

use params
use averager

implicit none
integer::n,ns,t,norm,iq,ii
integer,dimension(ns)::nx
real(8),allocatable,dimension(:,:,:)::y,z
real(8),allocatable,dimension(:,:,:)::pt,ps,ptr,mm,mmr,ts,tt,ttr,hs,ht,htr,s
real(8),allocatable,dimension(:,:,:)::ptmod,psmod,ptrmod
real(8),allocatable,dimension(:,:,:)::v,w,vb2b,wb2b,rho,ted,vc,wc,u,c,va,wa,vr,wr
real(8),allocatable,dimension(:,:,:)::vu,vv,vw,wu,wv,ww,vm,wm
real(8),allocatable,dimension(:,:,:)::alpha,beta,gamma,gamma1,radius,angle,angleo
real(8),allocatable,dimension(:,:,:)::v_a,vaa,vca,vra,w_a,waa,wca,wra
real(8),allocatable,dimension(:,:,:)::teda,mma,mmra,vb2ba,wb2ba,psa,pta,ptr
real(8),allocatable,dimension(:,:,:)::psamod,ptamod,ptramod
real(8),allocatable,dimension(:,:,:)::hsa,hta,htra,tsa,tta,ttra
real(8),allocatable,dimension(:,:,:)::alphaa,betaa,gammaa,gamma1a
real(8),allocatable,dimension(:,:,:)::rhoa,ya,za,sp,sa,ua,rad,anga,angao
real(8)::z_r_avg,z_s_avg,eta_s_avg,eta_r_avg,eta_st_avg,eta_sti_avg
real(8)::dhs1,dhs2,dhs3,tr
real(8)::vvec_i,num,den,ps3,ps4,ps5,t3,t4,t5,t4s,t5s,t5s1
real(8),dimension(ns,t)::ps_max,ps_min,pt_max,pt_min,ptr_max,ptr_min
real(8),dimension(ns)::av_p_max,av_pt_max,av_ptr_max
real(8),dimension(ns)::av_p_min,av_pt_min,av_ptr_min
real(8)::integ
real(8),dimension(t)::norm_b_pos

character(len=80)::indir,outdir,time,infile,feff,outfile,dum_string,fout1,fout2
character(len=4)::ext,nis
character(len=40)::cmd
character(len=20)::bladepos,stepn
character(len=1500)::superstring
tr=float(t)
allocate(y(n,ns,t),z(n,ns,t),pt(n,ns,t),ps(n,ns,t),ptr(n,ns,t),stat=iq)
allocate(mmm(n,ns,t),mmr(n,ns,t),ts(n,ns,t),tt(n,ns,t),ttr(n,ns,t),stat=iq)
allocate(hs(n,ns,t),ht(n,ns,t),htr(n,ns,t),s(n,ns,t),stat=iq)
allocate(v(n,ns,t),w(n,ns,t),vb2b(n,ns,t),wb2b(n,ns,t),rho(n,ns,t),stat=iq)

```

```

allocate(ted(n,ns,t),vc(n,ns,t),wc(n,ns,t),u(n,ns,t),c(n,ns,t),stat=iq)
allocate(va(n,ns,t),wa(n,ns,t),vr(n,ns,t),wr(n,ns,t),stat=iq)
allocate(vu(n,ns,t),vv(n,ns,t),vw(n,ns,t),wu(n,ns,t),wv(n,ns,t),ww(n,ns,t),stat=iq)
allocate(vm(n,ns,t),wm(n,ns,t),alpha(n,ns,t),beta(n,ns,t),gamma(n,ns,t),stat=iq)
allocate(gamma1(n,ns,t),radius(n,ns,t),angle(n,ns,t),angleo(n,ns,t),stat=iq)
allocate(psmod(n,ns,t),ptmod(n,ns,t),ptrmod(n,ns,t),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"

allocate(v_a(n,ns),vaa(n,ns),vca(n,ns),vra(n,ns),w_a(n,ns),waa(n,ns),wca(n,ns),wra(n,ns),stat=iq)
allocate(teda(n,ns),mma(n,ns),mmra(n,ns),vb2ba(n,ns),wb2ba(n,ns),psa(n,ns),pta(n,ns),ptra(n,ns),stat=iq)
allocate(hsa(n,ns),hta(n,ns),htra(n,ns),tsa(n,ns),tta(n,ns),ttra(n,ns),stat=iq)
allocate(alphaa(n,ns),betaa(n,ns),gammaa(n,ns),gamma1a(n,ns),stat=iq)
allocate(rhoa(n,ns),sp(n,ns),sa(n,ns),ua(n,ns),rad(n,ns),stat=iq)
allocate(ya(n,ns),za(n,ns),anga(n,ns),angao(n,ns),stat=iq)
allocate(psamod(n,ns),ptamod(n,ns),ptramod(n,ns),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"

if(iq /= 0)stop "****Not Enough Memory****"

ext='.dat'
indir='Contours/conv/'
outdir='out_contours_conv/'

do j=1,ns
  write(nis,'(I1)')j+2
  nis=adjustl(nis)
  do k=1,t
    write(*,'(A30,1x,I3,1x,A10,I6,1x,A2,1x,I6)')'cont: reading station',j+2,'for time= ',k,'of',t
    write(time,'(I4)')k
    time=adjustl(time)
    infile=trim(indir)//'cont'//trim(nis)//'_'//trim(time)//trim(ext)
    infile=trim(adjustl(infile))

    open(unit=3,file=infile,status='unknown')
    do i=1,nx(j)
      read(3,*)y(i,j,k),z(i,j,k),ps(i,j,k),pt(i,j,k),ptr(i,j,k),&
        v(i,j,k),vu(i,j,k),vv(i,j,k),vw(i,j,k),va(i,j,k),vc(i,j,k),&
        vr(i,j,k),vm(i,j,k),w(i,j,k),wu(i,j,k),wv(i,j,k),ww(i,j,k),&
        wa(i,j,k),wc(i,j,k),wr(i,j,k),wm(i,j,k),mm(i,j,k),mmr(i,j,k),&
        ts(i,j,k),tt(i,j,k),ttr(i,j,k),rho(i,j,k),s(i,j,k),hs(i,j,k),&
        ht(i,j,k),alpha(i,j,k),beta(i,j,k),gamma(i,j,k)
        vb2b(i,j,k)=sqrt(va(i,j,k)**2+vc(i,j,k)**2)
        alpha(i,j,k)=acos(vc(i,j,k)/vb2b(i,j,k))
        !beta(i,j,k)=atan(wa(i,j,k)/wc(i,j,k)) !just in case
        gamma(i,j,k)=atan(vr(i,j,k)/va(i,j,k))
        gamma1(i,j,k)=asin(vr(i,j,k)/v(i,j,k))
        radius(i,j,k)=sqrt(y(i,j,k)**2+z(i,j,k)**2)
        u(i,j,k)=omega*radius(i,j,k)
        wc(i,j,k)=vc(i,j,k)-u(i,j,k)
        wa(i,j,k)=va(i,j,k)
        wr(i,j,k)=vr(i,j,k)
        wb2b(i,j,k)=sqrt(wa(i,j,k)**2+wc(i,j,k)**2)
        w(i,j,k)=sqrt(wb2b(i,j,k)**2+wr(i,j,k)**2)
        beta(i,j,k)=acos(wc(i,j,k)/wb2b(i,j,k))
        pt(i,j,k)=ps(i,j,k)+0.5*rho(i,j,k)*v(i,j,k)*v(i,j,k)
        ptr(i,j,k)=ps(i,j,k)+0.5*rho(i,j,k)*(w(i,j,k)*w(i,j,k)-u(i,j,k)*u(i,j,k))
        ht(i,j,k)=hs(i,j,k)+0.5*v(i,j,k)*v(i,j,k)
        htr(i,j,k)=hs(i,j,k)+0.5*(w(i,j,k)*w(i,j,k)-u(i,j,k)*u(i,j,k))
        c(i,j,k)=v(i,j,k)/mm(i,j,k)
        mmr(i,j,k)=w(i,j,k)/c(i,j,k)

        angle(i,j,k)=atan(y(i,j,k)/z(i,j,k))
      end do
    close(3)
    angleo=angle
  end do

```

```

end do

!convert Pressure to kPa, energy to kJ/kg
!correct pressure values
ps=ps+pcor;pt=pt+pcor;ptr=ptr+pcor
ps=ps/1000.0;pt=pt/1000.0;ptr=ptr/1000
s=s/1000.0;hs=hs/1000.0;ht=ht/1000.0;htr=htr/1000.0
ext='.dat'
cmd='mkdir '//outdir
call system(cmd)

do i=1,ns
do j=1,t
ps_min(i,j)=minval(ps(1:nx(i),i,j))
ps_max(i,j)=maxval(ps(1:nx(i),i,j))
pt_min(i,j)=minval(pt(1:nx(i),i,j))
pt_max(i,j)=maxval(pt(1:nx(i),i,j))
ptr_min(i,j)=minval(ptr(1:nx(i),i,j))
ptr_max(i,j)=maxval(ptr(1:nx(i),i,j))
end do
end do

if((norm<=0).or.(norm>3))then
psmod=ps
ptmod=pt
ptrmod=ptr
elseif(norm==1)then
do j=1,ns
do k=1,t
do i=1,n
psmod(i,j,k)=(ps(i,j,k)-ps_min(j,k))/(ps_max(j,k)-ps_min(j,k))
ptmod(i,j,k)=(pt(i,j,k)-pt_min(j,k))/(pt_max(j,k)-pt_min(j,k))
ptrmod(i,j,k)=(ptr(i,j,k)-ptr_min(j,k))/(ptr_max(j,k)-ptr_min(j,k))
end do
end do
end do

elseif(norm==2)then
psmod=ps/psavg(1)
ptmod=pt/ptavg(1)
ptrmod=ptr/ptravg(1)
elseif(norm==3)then
psmod=ps/psmax(1)
ptmod=pt/ptmax(1)
ptrmod=ptr/ptrmax(1)

endif

if(docont==.true.)then
do k=1,t
norm_b_pos(k)=k*dt/dpitch(1)
write(stepn,'(I5)')k
write(time,'(E10.3)')k*dt
write(bladepos,'(F15.5)')k*dt/dpitch(3)
outfile=trim(adjustl(outdir))//'cont_'//trim(adjustl(stepn))//trim(ext)
outfile=trim(adjustl(outfile))
open(unit=5,file=outfile,status='unknown')
write(5,*)'TITLE="URANS CFD Interstage Flow Data'//trim(time)//' '//trim(bladepos)//'"'
call get_supstringc(ver,norm,superstring)
write(5,'(A1500)')superstring
do j=1,ns
if(j==1)write(5,*)'ZONE T="Station 3 CFD-URANS"'
if(j==2)write(5,*)'ZONE T="Station 4 CFD-URANS"'
if(j==3)write(5,*)'ZONE T="Station 5 CFD-URANS"'
do ii=-1,1
do i=1,nx(j)
angle(i,j,k)=angleo(i,j,k)+float(ii)*dtheta(j+1)
y(i,j,k)=radius(i,j,k)*sin(angle(i,j,k))

```

```

z(i,j,k)=radius(i,j,k)*cos(angle(i,j,k))

write(5,500)y(i,j,k),z(i,j,k),ptmod(i,j,k),ptrmod(i,j,k),psmod(i,j,k),&
v(i,j,k),vc(i,j,k),vr(i,j,k),va(i,j,k),w(i,j,k),wc(i,j,k),&
wr(i,j,k),wa(i,j,k),u(i,j,k),&
gamma1(i,j,k)*r2d,alpha(i,j,k)*r2d,beta(i,j,k)*r2d,&
mm(i,j,k),mmr(i,j,k),ts(i,j,k),tt(i,j,k),ttr(i,j,k),&
cp,s(i,j,k),hs(i,j,k),ht(i,j,k),htr(i,j,k),rho(i,j,k),&
rho(i,j,k),cp,gamma(i,j,k)*r2d,cp,cp,cp,cp,cp,cp

end do
end do
end do
close(5)
end do
end if
do j=1,ns
do i=1,nx(j)
ya(i,j)=0;za(i,j)=0;psa(i,j)=0;pta(i,j)=0;ptr(i,j)=0
v_a(i,j)=0;vaa(i,j)=0;vra(i,j)=0;vca(i,j)=0
w_a(i,j)=0;waa(i,j)=0;wra(i,j)=0;wca(i,j)=0
tsa(i,j)=0;tta(i,j)=0;ttra(i,j)=0
hsa(i,j)=0;hta(i,j)=0;htra(i,j)=0
alphaa(i,j)=0;betaa(i,j)=0;gammaa(i,j)=0;gamma1a(i,j)=0
sa(i,j)=0;rhoa(i,j)=0;mma(i,j)=0;mmra(i,j)=0;
vb2ba(i,j)=0;wb2ba(i,j)=0;rad(i,j)=0;anga(i,j)=0.

do k=1,t
ya(i,j)=ya(i,j)+y(i,j,k)
za(i,j)=za(i,j)+z(i,j,k)
psa(i,j)=psa(i,j)+ps(i,j,k)
pta(i,j)=pta(i,j)+pt(i,j,k)
ptr(i,j)=ptr(i,j)+ptr(i,j,k)
v_a(i,j)=v_a(i,j)+v(i,j,k)
vaa(i,j)=vaa(i,j)+va(i,j,k)
vra(i,j)=vra(i,j)+vr(i,j,k)
vca(i,j)=vca(i,j)+vc(i,j,k)
w_a(i,j)=w_a(i,j)+w(i,j,k)
waa(i,j)=waa(i,j)+wa(i,j,k)
wra(i,j)=wra(i,j)+wr(i,j,k)
wca(i,j)=wca(i,j)+wc(i,j,k)
tsa(i,j)=tsa(i,j)+ts(i,j,k)
tta(i,j)=tta(i,j)+tt(i,j,k)
ttra(i,j)=ttra(i,j)+ttr(i,j,k)
hsa(i,j)=hsa(i,j)+hs(i,j,k)
hta(i,j)=hta(i,j)+ht(i,j,k)
htra(i,j)=htra(i,j)+htr(i,j,k)
alphaa(i,j)=alphaa(i,j)+alpha(i,j,k)
betaa(i,j)=betaa(i,j)+beta(i,j,k)
gammaa(i,j)=gammaa(i,j)+gamma(i,j,k)
gamma1a(i,j)=gamma1a(i,j)+gamma1(i,j,k)
sa(i,j)=sa(i,j)+s(i,j,k)
rhoa(i,j)=rhoa(i,j)+rho(i,j,k)
mma(i,j)=mma(i,j)+mm(i,j,k)
mmra(i,j)=mmra(i,j)+mmr(i,j,k)
teda(i,j)=teda(i,j)+ted(i,j,k)
vb2ba(i,j)=vb2ba(i,j)+vb2b(i,j,k)
wb2ba(i,j)=wb2ba(i,j)+wb2b(i,j,k)
rad(i,j)=rad(i,j)+radius(i,j,k)

end do

ya(i,j)=ya(i,j)/tr
za(i,j)=za(i,j)/tr
anga(i,j)=atan(ya(i,j)/za(i,j))
psa(i,j)=psa(i,j)/tr

```

```

pta(i,j)=pta(i,j)/tr
ptr(i,j)=ptr(i,j)/tr
v_a(i,j)=v_a(i,j)/tr
vaa(i,j)=vaa(i,j)/tr
vra(i,j)=vra(i,j)/tr
vca(i,j)=vca(i,j)/tr
w_a(i,j)=w_a(i,j)/tr
waa(i,j)=waa(i,j)/tr
wra(i,j)=wra(i,j)/tr
wca(i,j)=wca(i,j)/tr
tsa(i,j)=tsa(i,j)/tr
tta(i,j)=tta(i,j)/tr
ttra(i,j)=ttra(i,j)/tr
hsa(i,j)=hsa(i,j)/tr
hta(i,j)=hta(i,j)/tr
htra(i,j)=htra(i,j)/tr
alphaa(i,j)=alphaa(i,j)/tr
betaa(i,j)=betaa(i,j)/tr
gammaa(i,j)=gammaa(i,j)/tr
gamma1a(i,j)=gamma1a(i,j)/tr
sa(i,j)=sa(i,j)/tr
rhoa(i,j)=rhoa(i,j)/tr
mma(i,j)=mma(i,j)/tr
mmra(i,j)=mmra(i,j)/tr
teda(i,j)=teda(i,j)/tr
vb2ba(i,j)=vb2ba(i,j)/tr
wb2ba(i,j)=wb2ba(i,j)/tr
rad(i,j)=rad(i,j)/tr
ua(i,j)=rad(i,j)*omega
end do
end do
angao=anga
do i=1,ns
av_p_min(i)=minval(psa(1:nx(i),i))
av_pt_min(i)=minval(pta(1:nx(i),i))
av_ptr_min(i)=minval(ptra(1:nx(i),i))
av_p_max(i)=maxval(psa(1:nx(i),i))
av_pt_max(i)=maxval(pta(1:nx(i),i))
av_ptr_max(i)=maxval(ptra(1:nx(i),i))
end do

if((norm==0).or.(norm>3).or.(norm<0))then
psamod=psa
ptamod=pta
ptramod=ptra
elseif(norm==1)then
do j=1,ns
do i=1,n
psamod(i,j)=(psa(i,j)-av_p_min(j))/(av_p_max(j)-av_p_min(j))
ptamod(i,j)=(pta(i,j)-av_pt_min(j))/(av_pt_max(j)-av_pt_min(j))
ptramod(i,j)=(ptra(i,j)-av_ptr_min(j))/(av_ptr_max(j)-av_ptr_min(j))
end do
end do
elseif(norm==2)then
psamod=psa/psavg(1)
ptamod=pta/ptavg(1)
ptramod=ptra/ptravg(1)
elseif(norm==3)then
psamod=psa/psmax(1)
ptamod=pta/ptmax(1)
ptramod=ptra/ptrmax(1)

endif

outfile='avg_out/Contour_Averaged_'//trim(rpms)//trim(ext)
open(unit=6,file=outfile,status='unknown')
write(6,*),'TITLE="URANS CFD Interstage Flow Data Averaged"'
call get_supstringc(ver,norm,superstring)

```

```

write(6,'(A1500)')superstring

do j=1,ns
if(j==1)write(6,*)'ZONE T="Station 3 CFD-URANS AVERAGED"'
if(j==2)write(6,*)'ZONE T="Station 4 CFD-URANS AVERAGED"'
if(j==3)write(6,*)'ZONE T="Station 5 CFD-URANS AVERAGED"'
do ii=-1,1
do i=1,nx(j)
anga(i,j)=angao(i,j)+float(ii)*dtheta(j+1)
ya(i,j)=rad(i,j)*sin(anga(i,j))
za(i,j)=rad(i,j)*cos(anga(i,j))
write(6,500)ya(i,j),za(i,j),ptamod(i,j),ptramod(i,j),psamod(i,j),&
v_a(i,j),vca(i,j),vra(i,j),vaa(i,j),w_a(i,j),wca(i,j),&
wra(i,j),waa(i,j),ua(i,j),gammaa(i,j)*r2d,alphaa(i,j)*r2d,&
betaa(i,j)*r2d,mma(i,j),mmra(i,j),tsa(i,j),tta(i,j),&
ttra(i,j),cp,sa(i,j),hsa(i,j),hta(i,j),htra(i,j),&
rhoa(i,j),rhoa(i,j),cp,gammaa(i,j)*r2d,cp,cp,cp,cp,cp,cp
end do
end do
end do
close(6)

write(*,*)'Contour Analysis -Done'

500 format(2(F10.5,1x),3(F15.5,1x),24(F15.5,1x),6(F8.1,1x))
501 format(8(F15.5,1x))

deallocate(y,z,pt,ps,ptr,stat=iq)
deallocate(mm,mmr,ts,tt,ttr,stat=iq)
deallocate(hs,ht,htr,s,stat=iq)
deallocate(v,w,vb2b,wb2b,rho,stat=iq)
deallocate(ted,vc,wc,u,c,stat=iq)
deallocate(va,wa,vr,wr,stat=iq)
deallocate(vu,vv,vw,wu,wv,ww,stat=iq)
deallocate(vm,wm,alpha,beta,gamma,stat=iq)
deallocate(gamma1,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

deallocate(v_a,vaa,vca,vra,w_a,waa,wca,wra,stat=iq)
deallocate(teda,mma,mmra,vb2ba,wb2ba,psa,pta,ptr,stat=iq)
deallocate(hsa,hta,htra,tsa,tta,ttra,stat=iq)
deallocate(alphaa,betaa,gammaa,gamma1a,stat=iq)
deallocate(rhoa,sp,sa,ua,rad,stat=iq)
deallocate(ya,za,stat=iq)
deallocate(psmo,ptmo,ptrmo,psamo,ptamo,ptramo,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

end subroutine

subroutine contour_anal_aux(n,nx,ns,t,norm)

use params
use averager

implicit none
integer::n,ns,t,norm,iq,ii
integer,dimension(ns)::nx
real(8),allocatable,dimension(:,:,:)::y,z
real(8),allocatable,dimension(:,:,:)::pt,ps,ptr,mm,mmr,ts,tt,ttr,hs,ht,htr,s
real(8),allocatable,dimension(:,:,:)::v,w,vb2b,wb2b,rho,ted,vc,wc,u,c,va,wa,vr,wr
real(8),allocatable,dimension(:,:,:)::vu,vv,vw,wu,wv,ww,vm,wm
real(8),allocatable,dimension(:,:,:)::alpha,beta,gamma,gamma1,radius,angle,angleo
real(8),allocatable,dimension(:,:,:)::ptmod,psmod,ptrmod
real(8),dimension(ns,t)::ps_max,ps_min,pt_max,pt_min,ptr_max,ptr_min

```



```

real(8)::z_r_avg,z_s_avg,eta_s_avg,eta_r_avg,eta_st_avg,eta_sti_avg
real(8)::dhs1,dhs2,dhs3,tr
real(8)::vvec_i,num,den,ps3,ps4,ps5,t3,t4,t5,t4s,t5s,t5s1
real(8)::integ
character(len=80)::indir,outdir,time,infile,feff,outfile,dum_string,fout1,fout2
character(len=4)::ext,nis
character(len=40)::cmd
character(len=20)::bladepos,stepn
character(len=1500)::superstring
tr=float(t)
allocate(y(n,ns,t),z(n,ns,t),pt(n,ns,t),ps(n,ns,t),ptr(n,ns,t),stat=iq)
allocate(mm(n,ns,t),mmr(n,ns,t),ts(n,ns,t),tt(n,ns,t),ttr(n,ns,t),stat=iq)
allocate(hs(n,ns,t),ht(n,ns,t),htr(n,ns,t),s(n,ns,t),stat=iq)
allocate(v(n,ns,t),w(n,ns,t),vb2b(n,ns,t),wb2b(n,ns,t),rho(n,ns,t),stat=iq)
allocate(ted(n,ns,t),vc(n,ns,t),wc(n,ns,t),u(n,ns,t),c(n,ns,t),stat=iq)
allocate(va(n,ns,t),wa(n,ns,t),vr(n,ns,t),wr(n,ns,t),stat=iq)
allocate(vu(n,ns,t),vv(n,ns,t),vw(n,ns,t),wu(n,ns,t),wv(n,ns,t),ww(n,ns,t),stat=iq)
allocate(vm(n,ns,t),wm(n,ns,t),alpha(n,ns,t),beta(n,ns,t),gamma(n,ns,t),stat=iq)
allocate(gamma1(n,ns,t),radius(n,ns,t),angle(n,ns,t),angleo(n,ns,t),stat=iq)
allocate(ptmod(n,ns,t),psmod(n,ns,t),ptrmod(n,ns,t),stat=iq)
if(iq /= 0)stop "****Not Enough Memory****"

ext='.dat'
indir='Contours/aux/'
outdir='out_contours_aux/'

do j=1,ns
write(nis,'(I1)')j+2
nis=adjustl(nis)
do k=1,t
write(*,'(A40,1x,I3,1x,A10,I6,1x,A2,1x,I6)')'aux cont: reading station',j+2,'for time= ',k,'of',t
write(time,'(I4)')k
time=adjustl(time)
infile=trim(indir)//'cont'//trim(nis)//'_'//trim(time)//trim(ext)
infile=trim(adjustl(infile))
open(unit=3,file=infile,status='unknown')
do i=1,nx(j)
read(3,*)y(i,j,k),z(i,j,k),ps(i,j,k),pt(i,j,k),ptr(i,j,k),&
v(i,j,k),vu(i,j,k),vv(i,j,k),vw(i,j,k),va(i,j,k),vc(i,j,k),&
vr(i,j,k),vm(i,j,k),w(i,j,k),wu(i,j,k),wv(i,j,k),ww(i,j,k),&
wa(i,j,k),wc(i,j,k),wr(i,j,k),wm(i,j,k),mmr(i,j,k),&
ts(i,j,k),tt(i,j,k),ttr(i,j,k),rho(i,j,k),s(i,j,k),hs(i,j,k),&
ht(i,j,k),alpha(i,j,k),beta(i,j,k),gamma(i,j,k)
vb2b(i,j,k)=sqrt(va(i,j,k)**2+vc(i,j,k)**2)
alpha(i,j,k)=acos(vc(i,j,k)/vb2b(i,j,k))
!beta(i,j,k)=atan(wa(i,j,k)/wc(i,j,k)) !just in case
gamma(i,j,k)=atan(vr(i,j,k)/va(i,j,k))
gamma1(i,j,k)=asin(vr(i,j,k)/v(i,j,k))
radius(i,j,k)=sqrt(y(i,j,k)**2+z(i,j,k)**2)
u(i,j,k)=omega*radius(i,j,k)
wc(i,j,k)=vc(i,j,k)-u(i,j,k)
wa(i,j,k)=va(i,j,k)
wr(i,j,k)=vr(i,j,k)
wb2b(i,j,k)=sqrt(wa(i,j,k)**2+wc(i,j,k)**2)
w(i,j,k)=sqrt(wb2b(i,j,k)**2+wr(i,j,k)**2)
beta(i,j,k)=acos(wc(i,j,k)/wb2b(i,j,k))
pt(i,j,k)=ps(i,j,k)+0.5*rho(i,j,k)*v(i,j,k)*v(i,j,k)
ptr(i,j,k)=ps(i,j,k)+0.5*rho(i,j,k)*(w(i,j,k)*w(i,j,k)-u(i,j,k)*u(i,j,k))
ht(i,j,k)=hs(i,j,k)+0.5*v(i,j,k)*v(i,j,k)
htr(i,j,k)=hs(i,j,k)+0.5*(w(i,j,k)*w(i,j,k)-u(i,j,k)*u(i,j,k))
c(i,j,k)=v(i,j,k)/mm(i,j,k)
mmr(i,j,k)=w(i,j,k)/c(i,j,k)

angle(i,j,k)=atan(y(i,j,k)/z(i,j,k))
end do
close(3)
angleo=angle
end do

```

```

end do
ps=ps+pcor;pt=pt+pcor;ptr=ptr+pcor
ps=ps/1000.0;pt=pt/1000.0;ptr=ptr/1000
s=s/1000.0;hs=hs/1000.0;ht=ht/1000.0;htr=htr/1000.0

do i=1,ns
do j=1,t
ps_min(i,j)=minval(ps(1:n,i,j))
ps_max(i,j)=maxval(ps(1:n,i,j))
pt_min(i,j)=minval(pt(1:n,i,j))
pt_max(i,j)=maxval(pt(1:n,i,j))
ptr_min(i,j)=minval(ptr(1:n,i,j))
ptr_max(i,j)=maxval(ptr(1:n,i,j))
end do
end do

if((norm==0).or.(norm>3).or.(norm<0))then
psmod=ps
ptmod=pt
ptrmod=ptr
elseif(norm==1)then
do j=1,ns
do k=1,t
do i=1,n
psmod(i,j,k)=(ps(i,j,k)-ps_min(j,k))/(ps_max(j,k)-ps_min(j,k))
ptmod(i,j,k)=(pt(i,j,k)-pt_min(j,k))/(pt_max(j,k)-pt_min(j,k))
ptrmod(i,j,k)=(ptr(i,j,k)-ptr_min(j,k))/(ptr_max(j,k)-ptr_min(j,k))
end do
end do
end do

elseif(norm==2)then
psmod=ps/psavg(1)
ptmod=pt/ptavg(1)
ptrmod=ptr/ptravg(1)
elseif(norm==3)then
psmod=ps/psmax(1)
ptmod=pt/ptmax(1)
ptrmod=ptr/ptrmax(1)

endif

ext='.dat'
cmd='mkdir '//outdir
call system(cmd)
do k=1,t
write(stepn,'(I5)')k
write(time,'(E10.3)')k*dt
write(bladepos,'(F15.5)')k*dt/dpitch(3)
outfile=trim(adjustl(outdir))//'cont_'//trim(adjustl(stepn))//trim(ext)
outfile=trim(adjustl(outfile))
open(unit=5,file=outfile,status='unknown')
write(5,*)'TITLE="URANS CFD Interstage Flow Data'//trim(time)//' '//trim(bladepos)//''
call get_supstringc(ver,norm,superstring)
write(5,'(A1500)')superstring
do j=1,ns
if(j==1)write(5,*)'ZONE T="Station 3 CFD-URANS"'
if(j==2)write(5,*)'ZONE T="Station 4 CFD-URANS"'
if(j==3)write(5,*)'ZONE T="Station 5 CFD-URANS"'
do ii=-1,1
do i=1,nx(j)
angle(i,j,k)=angleo(i,j,k)+float(ii)*dtheta(j+1)
y(i,j,k)=radius(i,j,k)*sin(angle(i,j,k))
z(i,j,k)=radius(i,j,k)*cos(angle(i,j,k))
write(5,500)y(i,j,k),z(i,j,k),ptmod(i,j,k),ptrmod(i,j,k),psmod(i,j,k),&
v(i,j,k),vc(i,j,k),vr(i,j,k),va(i,j,k),w(i,j,k),wc(i,j,k),&
wr(i,j,k),wa(i,j,k),u(i,j,k),&
gamma(i,j,k)*r2d,alpha(i,j,k)*r2d,beta(i,j,k)*r2d,&

```

```

mm(i,j,k),mmr(i,j,k),ts(i,j,k),tt(i,j,k),ttr(i,j,k),&
cp,s(i,j,k),hs(i,j,k),ht(i,j,k),htr(i,j,k),rho(i,j,k),&
rho(i,j,k),cp,gamma(i,j,k)*r2d,cp,cp,cp,cp,cp,cp

end do
end do
end do
close(5)
end do

write(*,*)'Contour Aux Analysis -Done'

500  format(2(F10.5,1x),3(F15.5,1x),24(F15.5,1x),6(F8.1,1x))
501  format(8(F15.5,1x))

deallocate(y,z,pt,ps,ptr,stat=iq)
deallocate(mm,mmr,ts,tt,ttr,stat=iq)
deallocate(hs,ht,htr,s,stat=iq)
deallocate(v,w,vb2b,wb2b,rho,stat=iq)
deallocate(ted,vc,wc,u,c,stat=iq)
deallocate(va,wa,vr,wr,stat=iq)
deallocate(vu,vv,vw,wu,wv,ww,stat=iq)
deallocate(vm,wm,alpha,beta,gamma,stat=iq)
deallocate(gamma1,stat=iq)
deallocate(ptmod,psmod,ptrmod,stat=iq)
if(iq /= 0)stop "****Issues Deallocating****"

end subroutine

```

APPENDIX B

CFX FORTRAN USER FUNCTIONS

B.1 Intermittency Function

```

#include "cfx5ext.h"
dllexport(user_gam)
    SUBROUTINE gam_unsteady(
        & NLOC,NRET,NARG,RET,ARGS,CRESLT,CZ,DZ,IZ,LZ,RZ)
C
C .....
C
C -----
C      Argument list
C -----
C
C      INTEGER NLOC, NRET, NARG
C      CHARACTER CRESLT*(*)
C      REAL      RET(1:NLOC,1:NRET), ARGS(1:NLOC,1:NARG)
C
C      INTEGER IZ(*)
C      CHARACTER CZ(*)*(1)
C      DOUBLE PRECISION DZ(*)
C      LOGICAL LZ(*)
C      REAL RZ(*)
C
C .....
C
C -----
C      Executable statements
C -----
C
C-----
C      GAM = RET(1:NLOC,1)
C      X      = ARGS(1:NLOC,1)
C      Y      = ARGS(1:NLOC,2)
C-----
C
C---- Low level user routine
C      call gam_unsteady_SUB(nloc,ret(1,1),args(1,1),args(1,2),
C      + args(1,3),args(1,4))
C
C      CRESLT = 'GOOD'
C      END

```

```

SUBROUTINE gam_unsteady_sub(nloc,gam,x,y,t,visc)

integer::nloc,iloc
real::x1,pwr,t
real,dimension(nloc)::gam,x,y,r,ang,visc
real,dimension(nloc)::gmin,gmax,gavg,s,rex

real,parameter::uin=10.6,r1=0.5,r2=0.702
real,parameter::res=1.2e5,ree=2.3e5,ub=-3.76
real,parameter::pi=3.14159265359
real,parameter::coeff=100.0
real,parameter::c1=0.5,c2=0.86,c3=0.8,c4=0.92

real::indx,put,para

gam=0.0;r=0.0;s=0.0;rex=0.0;ang=0.

open(unit=3,file='gamma.dat',status='unknown')

do iloc=1,nloc
  put=coeff*(x(iloc)-ub*t)
  para=(1+cos(put))/2
  r(iloc)=sqrt(x(iloc)*x(iloc)+y(iloc)*y(iloc))
  ang(iloc)=atan(y(iloc)/x(iloc))
  s(iloc)=r(iloc)*ang(iloc)
  rex(iloc)=uin*s(iloc)/visc(iloc)

  pwr=((rex(iloc)-res)/(res-ree))**2
  pwr=-1.0*pwr
  gam(iloc)=1.

  gmax(iloc)=1.0-c1*exp(pwr)
  gmin(iloc)=c2*(1.0-exp(pwr))
  gavg(iloc)=c4*(1.0-c3*exp(pwr))
  gam(iloc)=para*(gmax(iloc)-gmin(iloc))+gmin(iloc)
!   if(r(iloc)>0.701)then
!     write(3,10)t,x(iloc),put,para,gmax(iloc),gmin(iloc),
! +   gavg(iloc),gam(iloc)
!   endif

```

```

        if(gam(iloc)<0.0)then
            gam(iloc)=0.0
        elseif(gam(iloc)>1.0)then
            gam(iloc)=0.0
        end if

        if(r(iloc)>=0.7015)then
            gam(iloc)=gam(iloc)
        else
            gam(iloc)=gam(iloc)*((r(iloc)-0.692)/(r2-0.692))
        end if

    end do
10    format(8(F8.5,1x))
end

```

B.2 Inlet Velocity/Wakes Function

```

#include "cfx5ext.h"
dllexport(user_gam)
    SUBROUTINE inlet (
        & NLOC,NRET,NARG,RET,ARGS,CRESLT,CZ,DZ,IZ,LZ,RZ)
C
C .....
C
C -----
C      Argument list
C -----
C
C      INTEGER NLOC, NRET, NARG
C      CHARACTER CRESLT*(*)
C      REAL      RET(1:NLOC,1:NRET), ARGS(1:NLOC,1:NARG)
C
C      INTEGER IZ(*)
C      CHARACTER CZ(*)*(1)
C      DOUBLE PRECISION DZ(*)
C      LOGICAL LZ(*)
C      REAL RZ(*)
C
C .....
C
C -----
C      Executable statements

```

```

C -----
C
C-----
C      GAM = RET(1:NLOC,1)
C      X      = ARGS(1:NLOC,1)
C      Y      = ARGS(1:NLOC,2)
C-----
C
C---- Low level user routine
      CALL inlet_SUB (NLOC,RET(1,1),ARGS(1,1),ARGS(1,2))
C
      CRESLT = 'GOOD'
      END

      SUBROUTINE inlet_SUB (NLOC,UIIN,X,T)
C
C .....
C
C -----
C      Local Variables
C -----
      INTEGER NLOC, ILOC
      real,dimension(nloc)::uin,x,ang
      real :: unom,t,x1,ub,Vy,Vx,para,coeffa
c      REAL      GAM(NLOC), X(NLOC),Y(NLOC),R(NLOC),ANG(NLOC)
c      REAL      S(NLOC),REX(NLOC)
      PARAMETER(Vy=10.6)
      PARAMETER(Vx=0.0)
      parameter(ub=-3.76)
      parameter(coeffa=100.0)
      uin=0.0
      x1=0.0

      do iloc=1,nloc
        ang(iloc)=coeffa*(x(iloc)-ub*t)
        para=(1+cos(ang(iloc)))/2
        uin(iloc)=Vy*para
      end do
      END

```